EVERY MONTH JANUARY 1991 £1.75

SINCLAIR QL WORLD

2 4 to 7 9 10 11 12 13 14 15
16 20 22 24 26 28 30 CENT. COL
32 35 38 44 47 50

BUY A QL!

Mix your own computer – we have the formula!

DIY Toolkit
CLIPBOARDING THE QL

Software File
HOME BUDGET
QL GENEALOGIST

ARCHIVE: THE ABSOLUTE BEGINNING

SINCLAIR
QL WORLD

Editor
Helen Armstrong

Production Controller
Jayne Penfold

Designer
Jeff Gurney

Advertising Sales
Jason Newman

Magazine Services
Sheila Baker

Advertising Production
Michelle Evans

Group Advertising Manager
Jean Dorza

Group Editor
John Taylor

Publishing Director
Ray Lewis

Managing Director
Peter Welham

Sinclair QL World
Panini House
116-120 Goswell Road
London EC1V 7QD
Telephone 071-490 7161
ISSN 026806X

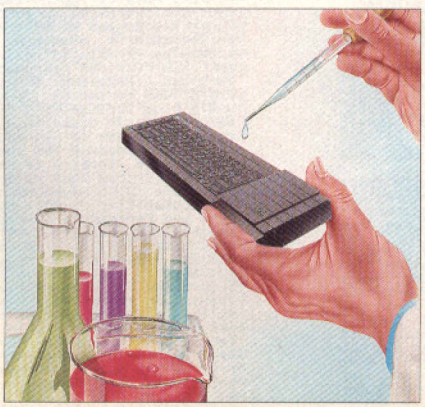
Unfortunately, we are no longer able to answer enquiries made by telephone. If you have any comments or difficulties, please write the The Editor, Open Channel, Trouble Shooter, or Psion Solutions. We will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies.
Back issues are available from the publisher price £2 U.K., £2.75 Europe. Overseas rates on request.
Published by Maxwell Specialist Magazines, A Division of MCPC Ltd., Sinclair QL World is distributed by SM Distribution.
Tel: 081-677 8111. Subscription information from: MSM Subscription Dept, Lazahold Ltd., PO Box 10, Roper St, Pallion Ind. Est., Sunderland SR4 4SN.
£21.00 U.K., £24:70 Europe, MiddleEast £25.80, Far East £27.60, Rest of World £26.20, U.S.A. $45.00.
Airmail rates available on request.
Typesetting by Ford Graphics, 8-10 Whitsbury Road, Fordingbridge, Hampshire. SP6 1BR. Tel: (0425) 655657
Printing by Cradley Print. (0384) 411411
Sinclair QL World is published on the third Thursday preceding cover date.

© COPYRIGHT
SINCLAIR QL WORLD – 1990

CONTENTS

JANUARY 1991

9 QL SCENE ● Minerva and Merz
10 OPEN CHANNEL ● The good guys
12 QL SCENE ● The new one from Digital Precision!
17 THE QL IN ITALY ● A report from the Italian scene
18 THE QL IN BRUSSELS ● A successful European Microfair
20 TROUBLESHOOTER ● Ideas for a new QL
24 PSION SOLUTIONS ● Quill and Abacus suggestions
26 BUYING FOR BEGINNERS ● How to buy a complete QL system
28 DIY TOOLKIT ● Clipboarding the QL
34 SOFTWARE FILE ● Home Budget update
35 SUPERBASIC ● The final part of Space Marauders
38 THE ABSOLUTE BEGINNING ● Archive from the start
42 SUBSCRIPTION INFORMATION
44 SOFTWARE FILE ● QL Genealogist
46 SOFTWARE FILE ● Vision Mixer
48 SOFTWARE FILE ● Aqua Vitae



NEXT MONTH

THE USER GUIDE GUIDE

Coming soon: a new look at the QL User Guide for new users.

PROGRAMMING IN C

The last part of our introduction to this advanced programming language.

QL SCENE

Minerva Snags

QL Users moving on to later rom versions of *Minerva* are reporting some problems running established QL software packages.

Minerva is currently on V1.82, reported to be the final version for the time being. It appears that so many upgrades and alterations to the original Minerva concept have now appeared that software publishers cannot keep up with – or, indeed, cover – all the differences between Minerva and the various Sinclair-vintage QL roms.

Said one user: "As far as I can see, the problem is not a matter of bugs in either Minerva or the software, even where bugs do exist, but that Minerva is effectively a new operating system which has moved away from existing operating systems for the QL and Thor."

QView made clear that they are aware of the problem earlier this year when they stated in advertising: "Compatibility can be a bit of a problem when your fundamental idea is to change everything in sight. We currently have over 400 customers and know of no problem for which there is no solution, either available now or soon."

However, some software suppliers report that Minerva users have been referred to them for 'bug fixes' when the software in question works correctly with Sinclair and Thor roms.

The fear is that users will become bewildered by apparent software bugs and suppliers will be swamped with requests for alterations or 'de-bugging' to meet Minerva specifications, interfering with their own work.

There may be an ominous precedent for this view from pre-Minerva days. Pundits noted supplier PDQL's former willingness to non-upgrade software adjustments, particularly printer-drivers, at an unrealistically low price, as one factor in its apparent financial breakdown. An observer said at the time: "John must be doing a lot of work and only charging £5. How can he cover his business expenses?"

Companies have already spent time establishing whether faults reported are the result of changing the operating system, rather than inherent in the software.

One suggested that QView should allow customers one free alternative version of their choice, now that the rate of change has reached a plateau, instead of their long-standing offer of one free upgrade.

Absolute Chaos

A company called Frachaos have published a catalogue of 'proprietry chaos products', including videos, books, shareware, fine art prints and commercial software, related to fractal images.

Computer artist and proprietor Jake Davies says: "Since the Equinox programme screened two years ago there has been a rapidly growing interest in fractals and chaos, not only in mathematics and the sciences, but also in entertainment and fashion." The Catalogue of Chaos is apparently the first of its kind in the UK and contains a number of fractal-related products not previously available in the UK.

Copies of the Catalogue of Chaos are available free from Frachaos, Higher Trengrove, Constantine, Falmouth, Cornwall TR11 5QR. Tel. (24 hours answerphone) 0326 40973.

Another Piece of Pie

Author Geoff Wicks has sent a correction to *3D Pie*, published in the November 1990 issue. "There is a programming howler in lines 3620 and 3820", he writes. "If 'seg' is always an integer, what is the point of INT(seg)/seg? The moral is, never do something in a complicated way when SuperBasic provides a simpler means." He suggests as alternatives:

```
3620 IF fink=4 AND seg MOD
 3 <> 1
3680 IF fink=120 AND seg
MOD 3 <> 1
```

Geoff adds: "The program can be compiled quite easily using *Turbo* or *QLiberator*. It also benefits from *Lightning Special Edition*, but standard *Lightning* does not support all the stipples. Users of Lightning SE should have only the text and graphics routines installed, or should switch off the Maths routines using _lngPREC 0,".

Merz Update

Jochen Merz Software has issued a new leaflet and price list. The current list includes the *QDII* editor V2.38 disk, £36.50, the new *QMenu* menu extension V1.02, giving 'very complex but easy-to-use menus', disk £10.50, *Thing and Eprom Manager* II V2.08, which now accepts command strings and a supplied channel ID for a log file, among others, mdv or disk £18, QLQ font editor for 24-pin printers, V1.12 disk, £20, *QSUP* replacement for *QSYS I* and *II*, V1.13 disk or mdv £26, new games *Diamonds* (a bit in the style of *Tetris* and 'even more of a challenge for anyone who likes playing *Brain Smasher*') disk or mdv £10.50, *Patience* disk £13.50, and classics *Brain Smasher* disk or mdv £10.50, *Arcanoid II* disk or mdv £10.50, *Firebirds* disk or mdv £10.50, *Ion Gold* and *Doppel Ion*, two German coin-op gambling simulators, disk £10.50.

Merz also supplies *QLiberator* compiler V3.13 disk or mdv £51, *QLoad/QRef* fast load and save, disk or mdv £46.50, *The Painter*, now also on the QL emulator for the ST, disk £29.50, *QIMI* internal mouse interface for any Atari-compatible mouse, £39.90, Mouse for QL or Atari ST, £26, and a keyboard interface for PC or XT, plus 'most switchable AT and XT keyboards', now with keyboard driver giving extra features on spare keys, at the new lower price of £41.

The new V3 of *QD* is ready for release. Improvements over *QD II* include cursor handling, and complex command strings. Eprom utilities or an assembler can be called directly from QD, used and returned to QD. Block functions have been extended and 'you can even stuff a marked word or line into the HOTKEY stuffer buffer', according to Merz. There are other new and useful facilities. V3 is the same price as QD II: £36.50. The upgrade charge from QD I is £18.50 and from QD II £10, including manual.

The *QL Emulator* for the Atari ST now has a better keyboard driver to level C-14, a new floppy disk driver, new parallel port driver, new serial port driver (all written by Tony Tebby) plus a new version of Tebby's *SuperToolkit II* and other features. Various small bugs appearing with hard drive use have been fixed, and error trapping with same extended. The QL Emulator MODES for the Atari ST costs £163 and allows the QL's advanced software and operating system to be used with the ST's modern hardware and keyboard. Extra manuals are available for £6. For more details and full price list, contact **Jochen Merz Software, Im Stillen Winkel 12, 4100 Duisberg 11, West Germany. Tel. (Germany) 0203 501274.**

OPEN CHANNEL

Open Channel is where you have the opportunity to voice your opinions in *Sinclair QL World*. Whether you want to ask for help with a technical problem, provide somebody with the answer, or just sound off about something which bothers you, write to: Open Channel, Sinclair QL World, 116/120 Goswell Road, London EC1V 7QD.

Vroom

I have been reading *QL World* and its predecessor *QL User* since the magazines inception in 1984 and I have never failed to find something of interest. I have had my AH version machine right from the beginning (mail order from Sinclair) and the only things that have gone wrong with it in that time are the mdvl_ula blowing up, requiring an expensive replacement, and the power socket coming adrift from the pcb, which I got an electronics hobbyist to repair for free.

I have had various problems with microdrive cartridges, having to throw away three in six years – but apart from this I have found mdv cartridges to be reliable in the most unlikely circumstances. For example, two years ago I used Quill (V2.0) to word process by GCSE computer studies project: ten pages of condensed A4 print with the line spacing on my printer slightly adjusted to get even more lines per page. I wrote it in about five chunks, and (probably unwisely) merged them. The resulting file took half an hour to save and 50K of cartridge space, but I suppose 10,000 words is a slightly excessive document to write on an unexpanded QL with microdrives only.

I lost the document once, not having enough mdvs to back it up onto – another unwise move – but was able to recover 70% of it using a program that read in the file a character at a time until the "bad or changed medium" error turned up. I then tidied up the file in Metacomco Ed, and imported the resulting ascii file into Quill. After some work to reformat the file, it was as good as ever. Better, actually, as the close scrutiny allowed me to tidy up my style and a lot of spelling mistakes.

Could I make a plea? About three years ago my brother bought me a copy of the Pyramide game Vroom! as a birthday present. I used it for light relief quite happily until the backup gave me a "bad or changed medium" message. Undaunted, I loaded up the backup program from the master cartridge and set it running. Everything went well until about three quarters of the way through the master cartridge a "bad or changed medium" message came up right in the middle of the important game code.

This was very annoying and as I don't have a cartridge doctor I was unable to salvage the code, leaving me with a scuppered game and an unhappy brother. Therefore, I am asking if anyone out there can help me get my copy working again. I would be most grateful.

I aim to expand my system in the next few months and so am looking for a cheap second-hand disk interface and dual drives as well.

Keep up the good work with *QL World*.

Euan Holton
7 Stoney Lane
Thatcham
Berks. RG13 4LH

Commodore

I hope you can solve a problem for me. I recently purchased a JS rom QL on the understanding that microdrive cartridges were freely available, which they are not now, as we know.

I have however got a Commodore 40/40 dual 5.25 in disk drive that I use with a CBM computer. I was wondering if it is possible to interface this drive with the QL. I do not see why not, and it seems a waste of a drive if I cannot use it.

I am a disabled person and I therefore cannot afford to buy the interface or the drives from any supplier in your magazine.

Could you let me know how I would go about the problem? I have not got the connections for the 40.40 drive as I do not have its user manual.

J G Wilkinson, Coxford
Southampton

Editor's comment: Fortunately, microdrives are coming back into service, as reported in August's QL World.

I do not know the disk drive you mean. In theory, there is no reason why it should not be serviceable with the QL, but in practice if the drive was custom-made for a Commodore machine, it may not be practicably convertible for use with a QL.

If the drive has the Commodore brand name, you may be able to find out about the connections from Commodore on 0628 770088. Ask for the advice about an old Commodore disk drive. If however it has a different brand name on it, you will have to talk to the manufacturers, whoever they are. They may do a double-take when you say you want to interface it to a QL, as many people don't know what a QL is. We receive countless press releases from companies who think it's a type of PC clone.

However, having said that, even with all the information it would be inadvisable to undertake to try to build a disk drive interface unless you have considerable experience in building computer electronics.

Without that experience, there is no really cheap path to disk drives. Disk has always been a big jump up from tape storage, and the expense is the main reason.

Editor's notebook

Our second cover booklet will be followed up next month with a first-steps guide to using *Abacus*, and in the planning channel for a future issue - maybe even February - is that oft-requested item, a user's guide to the *QL User Guide*. That most essential of documents has been plagued from the start by misprints and errors. We aim to nail the worst of them and make 'the book' easier for new owners and beginner-programmers to use.

Digital Precision have at last released their newly-developed blockbuster word processor (see *QL Scene*, Page 12). Freddy Vachha assures me that, alongside all the fragrently-named Prefection's virtues and up-the-minute facilities, it has also been designed for the *Quill* user to slip into without tears. We hope to have a full review of that in the near future.

Due to circumstances beyond my control, some readers and subscribers did not get the April 1990 issue of *QL World*. I am now in a position to assist, while stocks last. Please send a large SAE with at least 22p postage (inland UK) or sufficient IRCs for overseas postage, and your subscriber number, if you want a copy of April 1990. Non-subscribers please send £2 per copy to QL World as usual.

Happy Christmas and a peaceful New Year.

Woodlice

I ordered a QL when they were first advertised as the ideal machine for a small business. Eventually I received an AH and went through all kinds of self-torture until it dawned on me that the errors were not of my own making. Over the years, with the addition of a printer (Star Delta 10 now changed to LC 10), Trump Card, disk drives and a Schön keyboard as extra hardware and *Flashback* (especially SE), *Lightning* and *Qram* among my software, the AH served me very well in my work as a Methodist minister.

A Quanta advert in *QL World* a few months ago for cheap modems tempted me and I fell. By the time I wrote they had run out of the auto-dial version. Was this a hint from above? I didn't listen too long, in case it was, but rang Sector Software who supplied me very quickly and suggested I might like the cut version of Qualsoft *Terminal* as well.

What amazed me was that I set it all up and it worked! Usually, I am about as good at dealing with bits for computers, soft or hard, as I am at looking at a Hebrew text – and at least I know that's meant to go backwards. My only successful piece of programming is one that lets me set the clock when I boot up and that I only did because I couldn't cope with all the parameters for SDATE. I printed out the manual for Qualsoft Terminal, did what I thought it said and it happened! I found myself in touch with Prestel. Much encouraged, I tried the TF Services bulletin board QBBS and got through!

The Tandata modem and Qualsoft's Terminal have opened up a whole new dimension for me and my computer. At first, there were lots of words I didn't understand. I have found Tony Firshman, the QBBS co-sysop, helpful and patient in explaining. Since then I have made contact with a number of bulletin boards which cater for a wide variety of interests. It's fascinating as well to read messages from around the world thanks to FIDO networks. It is useful to be able to leave queries on the QBBS or QLeaps.

The first thing I noticed with QBBS was that it was "now running under Minerva", and I had no idea what it was. However, the QL World review and Tony Firshman's experience encouraged me to make contact with QView Mega Corporation. Minerva is easy to fit . . . well, I still need to sit down when I think of fitting the Schön keyboard. I received a friendly letter back from Stuart McKnight at QView saying that there were a few things that would need to be done but nothing more than if I wanted to upgrade from the AH rom to the JM or JS (but that was why I had never upgraded).

In the meantime, as one is always being given the impression that the world is running out of QLs, I had bought a JS version. Taking a deep breath, I sent off for two rather large woodlice clinging to a bed of foam in a Jiffy bag. The JS was easy to do and immediately it transformed the machine. I don't use the QL to its limits but the difference was at once obvious and the whole machine has become more reliable. QView kindly offered to do the surgery necessary for the AH brain transplant and I arranged to take the AH machine to the Quanta Mid-Anglia Sub-group.

I must confess that I had reservations about mixing with a lot of QL enthusiasts. That's why I only recently joined Quanta. I found a group of enthusiasts whose words were sometimes above me, yes, but also a very friendly group of people, who were eager to make me welcome and be of any help they could.

Jonathan Oakley successfully evangelised my QL, with a little help from other members, which made me glad he had done it and not me! During the evening, Tony Tebby appeared and gave a lively demonstration of how near QPAC 2 was to being ready for sale. The whole atmosphere of the meeting was very positive and I found myself encouraged about the future of the QL. It was worth the journey from Watford.

My plans for the future include the Astracom modem, *Text 87*. I suppose I should look at a hard disc.

The point of this rather rambling contribution is to say, thank you to QL World for keeping me in touch with the QL world and thank you to all the people and organisations mentioned above and others who have helped make the QL not only an increasingly useful help in my work, but also a pleasure.

Roy Jackson
Watford
Herts

And thank you for supporting the QL. It works both ways.

Lobster

I work my computers as tools of my technology through available software by courtesy of an aging brain from the days of slide rules and log tables. There is no time nor inclination to understand, learn, assimilate and use much of the stuff featured in *QL World* and games are for the young and/or idle.

Taskmaster is tailor-made for my reducing braincells and I am never without it. It works for me and not me for it. The article on *Taskforce* is beyond my comprehension, leaving me with a filed (like *Ftidy*) program that I cannot work, because the vanity of your magazine does not consider that anyone using computers is not in with the workings of the language and cannot chat away in jargonese.

I have a copy of *Cadette* and find the operating manual worse than an arabic doctor's prescription for cod's wallop. Over the years I have passed thousands of tonnes of British sterling pounds through this QL with the help of the three Psions. All generated by escaping from the old hard ways, yet unable to absorb much of the new. I work hard at it.

If, by chance, there is anyone there who finds *Wordstar* and *Supercalc* 70% overloaded with unrequired abilities and is yet blissfully happy with Psion at the same game, could you get him to communicate about Taskforce, Cadette and the like, as if they were meant for money-earning users and not the lad up the hill part way through his third year in computer science.

To end, there is a lot to be said for the divisibility of the bits and pieces, like a keyboard gets an *Expanderam* plus a disk interface plus drive plus Mirovitec plus a printer. The incoming travelling computer is overcharged with facilities that cost money I for one do not wish to spend. Us older young are at home with QL, though my younger colleagues and friends would have me and it in the breaker's yard perhaps.

I remain hopeful.

Roy Myers
Haddington
East Lothian

Editor's comment: The point raised about 'jargon' is an interesting one, Computer people are often criticised for 'talking in jargon' and indeed they do . . . but not all the time. They talk in gobbledygook as well, and umgalahgalah (especially PC system salespeople), but the incomprehensible gibbering that serious software people talk when they are communicating serious business is not jargon. It's just language: if you don't understand it, it's because you haven't learned it yet. There isn't an easier way of saying it.

To quote a computer tutor: "a cursor is a cursor, you can't say "the little green thing that you push round the screen" for the rest of your life." You find out what it is by asking and by using it.

People sometimes ask why we don't publish glossaries. It's because glossaries are a waste of your £1.50. There are dozens of low-price computing dictionaries on the market; for specialist QL terms, refer to your User's Handbook. The more obscure terms can't be explained by dictionary definitions. It would be like trying to grow and cook a lobster by knowing that it was a blue crustacean with pincers. You have to wrestle with them (and lobsters).

You are underestimating your brain cells. Come on, admit it: you don't have time, you have too much work to do, you need software that will work for you as quickly as possible without knowing anything about what goes on inside. But don't blame the grey matter. Some people start computing at 80. The breakers' yard only takes volunteers!

Having said that, I couldn't agree more (in general, not knowing Cadette) about manuals and hope somebody can help.

PERFECTION
the new one from DP

To answer all your questions: yes, Freddy Vaccha at Digital Precision has been working on something big, and 'Fred's Big New Program' is ready for release. It is called *Perfection*. The same team – headed by Freddy and Steve Sutton – that brought out DP classics like *Lightning Special Edition* and *PC Conqueror*, have spent two years designing this new word processor to bring the best modern features to the QL, while retaining the familiarity of *Quill*. DP say that Perfection, which also has database applications, will "revolutionise the way you use your QL."

QL Quill could be mastered in a few minutes thanks to its simple menu system, but it lacks many commands now expected from a word processor. Attempts to accelerate it by 'patching' did not increase the speed much.

Intuitive

There is, however, no getting away from the fact that the majority of QL owners still use Quill as their main program.

So here is a new program which, say DP, can be mastered in a couple of minutes, and uses all the keys that you would intuitively expect to use – no cumbersome re-learning of keystrokes. It is menu-driven with multiple page menus, so there is no need to memorise anything, or ever refer to the manual. Perfection, say DP, is even quicker to master than Quill (those users who use other word processors, even the popular *Wordstar*, will understand why we take this so seriously). This is a word processor "designed for absolute beginners and advanced users . . . for letters and documents of a few pages, and articles, journals, magazines, books, theses or manuscripts hundreds of pages long".

Perfection is also designed to be user-configurable and printer-friendly but many users will never have to configure it. It is supplied working with Epson-compatibles and most non-compatibles. If you have a different printer, it can use your existing configured Quill printer data file, automatically, if you want it to.

Perfection has full on-screen character mode indication – bold, underline, italics, and even sub- and super-scripts. Other type modes – depending on your printer – such as switching fonts, pitches, NLQ/draft mode, proportionality, double-strike, etc. are indicated by strip colour combinations on screen, like the most advanced PC word processors. You can also define your own on-screen modes – even combinations of attributes are permitted.

As if clarity, flexibility and simplicity were not enough, Perfection in true DP style is the fastest word processor for the QL. DP's own tests check it in at 'over a hundred times faster than Quill on many common operations', dozens of times faster than Quill on most, and a minimum of five times faster at its slowest. Most to the point, DP admit it is many times faster than their own *Editor* text-handler.

Existing

But even more to the point, existing users of Quill and other QL word processors will find that Perfection can load existing saved _doc, _lis or Ascii files directly, with no conversion process required. This – together with the automatic reading of existing printer driver data – takes the terror out of moving to an advanced new system. Perfection files are also usable with PC, ST and Amiga word processors.

Perfection includes a multi-function configurator that allows you to tailor your own version if you like. Practically everything that can be set at run-time is also pre-configurable. If you no longer need the menus to be visible all the time, you can toggle them off, or configure it so that the menu is not visible. Or you can opt to bypass the menu system, and use keypress commands.

Multitask

Perfection multitasks natively, so that you can run multiple copies, as well as other software at the same time. Even with one copy running, you can take a 'snapshot', and view that as you edit a different part of the document. You can set up macros so that making a glossary is easy. You can have any number of blocks defined in the document. You can undo/edit attribute changes with a single keypress.

Dodging human error is a part of the *Perfection* design philosophy. There is an Undo option, and you can also ESCape from any command. With right-justification, Perfection adds pseudo-spaces to pad out the line, so if you accidentally right-justify tab or column material, a simple left-justify will get it back to its exact original state. Most word processors do not distinguish between keyed spaces and pseudo-spaces.

For those who want it, Perfection can be bundled with a high-speed dedicated spelling checker. This combination is appropriately called *Perfection Plus*. There are two levels of dictionary (you get both) – a larger one of 225,000 words (slightly less if your system is non-Trumpcard) and a compact one: use the latter if you are short of memory, or when your document is really huge. You can add new words to the dictionary and also create new dictionaries. You can check as you type as well as check saved files, and, best of all, you can spell-check interactively from any point in the document to another.

DP admits that there is one area where *The Editor Special Edition* remains master: the editing of 'non-print' data, handling the Ascii character set from codes 0 to 255. If you are a technical user and do not have either Editor Special Edition or Perfection, you can buy both programs together (they coexist, work simultaneously and have fully-compatible file formats); you will then get Editor Special Edition at half price – an impressive bargain.

Database

As *Perfection* is a completely new development, it is not available as an upgrade. However, existing Editor owners can claim a 5% reduction when ordering it.

An unexpected angle: a database can Store, Retrieve and Manipulate information rapidly. Perfection makes a good database system on these lines. Forward and backward search takes under ten seconds in the worst case with a 750 page document (a typical search time is 50 milliseconds). Cursor navigation is fast with an accelerating scroll rate if you want. There are macros, programmability and more for the more advanced user. It can

Continued on next page.

QL SCENE

Continued from page 12

even access existing *Archive* export files.

If you want full desktop publishing capabilities (the use of fonts that your printer does not possess, and graphics) you can link Perfection with *Professional Publisher*. Use Perfection for creating, editing and manipulating, and send the result to Pro Publisher. Bold, italics, underline, superscript, subscript etc. can be retained, and it will be possible to link graphics and pictures into your word processor files.

Perfection runs on any expanded QL with a disk drive. It costs £79.95 including integrated printer drivers, ancillary programs and documentation. Perfection Plus, with the dedicated Spelling checker and dictionaries, costs £119.95.

Those who want the spelling checker later can get it on Digital Precision's usual upgrade terms.

Other design elements of Perfection design include lazy screen (you scroll only the line your cursor is on, instead of updating the screen all the time), hashing and lazy attributes (where you do a jump of hundreds of pages, the destination area is displayed immediately, and the attributes 'catch up' a second later). A garbage-job runs in the background, so the internal tables are always in a perfect state. Both lazy screen and cursor acceleration are user-configurable.

Perfection has other tricks. For instance, when searching, it will look for rare letters before common ones, cutting down search time.

For advanced users, Perfection has full programmability with the ability to save and re-execute programs. There are over a hundred commands. File formats contain one short header with margin and tab, etc., data for that document and then whatever you type in. Changes of attribute are stored as invisible control characters in the file. If you opt to Export, the header is suppressed, to enable its direct use as a programming or technical front end, or to allow its output to be read in by another word processor.

Perfection, says DP, will blow your socks off.

Professional Publisher Toolbox

For *Professional Publisher* users comes a useful addition that not only supplies several man years' worth of beautiful high definition fonts – including familiar types like Roman and Universal – but contains many smaller fonts, more clipart and programs to load Sector Software clipart, filter text before importing into Professional Publisher, save parts of Professional Publisher pages as screens (for importing into any graphic program – like *Eye-Q* – or manipulating via SuperBasic, etc.) Excellent value at £29.95.

QL users wanting a state-of-the-art desktop publishing system can get the whole Professional Publisher system (*Professional Publisher, Toolbox, Font Enlarger* and *grafiX* printer driver) for £149.80 less 20% ie £119.85. Adding Eye-Q for advanced graphics makes the price up to £189.75 less 25%, ie £142.

Digital Precision have a special offer on the whole system (including PP, Toolbox, Font Enlarger, grafiX, Eye-Q) plus Perfection and Special Edition Editor (total list price £319.65) for £199 (£239 with Spelling Checker), a discount of almost 40%.

Turbo Price Slash

Digital Precision have cut the price of the flagship *SuperBasic* compiler *Turbo* by 20% to £79.95.

Digital Precision Ltd., 222 The Avenue, Chingford, London E4 9SE. Tel: 081 527 5493.

All Formats

The first All Formats Computer Show is on 2 February at the New Horticultural Show, Westminster, London. For further details and stand prices contact **John Riding on 0225 447453.**

PD Library

R. Dunnett's QL Users' Bulletin Board in Essex is being extended into a new QL public domain software library. To be know as Qubbesoft (pronounced 'cubesoft') P/D, the library, which is modest in size at present, makes programs available on 3.5 inch disk only (single or double-sided). Each disk contains ten programs plus a boot to run the Main Menu Handler. Programs will run from the main menu handler unless otherwise stated.

The contents of each disk are free apart from a copying fee of 50p. Software can be obtained either by sending sufficient disks with the fee of 50p per disk, in a jiffy bag with a return address label and return postage or IRCs or state which disks you require and enclose total fee of £1 per unbranded disk (50p for the disk and 50p for copying), address label and sufficient return postage or meet Mr. Dunnett at a Quanta workshop where he will be available to copy disks.

All cheques or money orders should be made payable to K. Dunnett (sic) and crossed.

Mr. Dunnett is currently organising a catalogue of the Library which will be free to senders of a large stamped self-addressed envelope, or through the bulletin board Qubbe. He is also looking for further contributions to the library, which is open to all QL users.

Contact **R. Dunnett at 38 Brunwin Rd., Rayne, Braintree, Essex CM7 5BU or through the Qubbe bulletin board on 0376 47852** (Viewdata-style ringback type).

The last three lines of the program accompanying Neil Gordon's *One Man's System* in the November 1990 QL World were omitted. They are:

```
4200 PAUSE -1
4210 displayalldata
4220 END DEFine
```

Errata

Author Ian Thompson has provided two corrections for his program *Chemistry* after reader James McGreehan pointed out a discrepancy. He writes: "Line 690 is indeed missing from the listing, and line 1690 should be used, as James suggests. Line 690 definitely exists in my copy, but may not have done when I sent in the original some time ago.

"As for the mistake in the Periodic Tables – oops. The correct solution is as follows:

"Alter line 870 so that the last figure is 32 (not 33). This then prints Ge 32 correctly in column 4A.

"Alter line 800 so that the last figure is 34 (not 35). This removes the duplicated Kr 36 in column 7A."

Congratulations to the following winners in the MSM October subscription renewal draw:
J Shooling of Windsor wins £150.
D G Ulph of Norwich wins £50.
T F Yarney of Winchester wins £50.
When subscribers renew, their names will automatically be entered in the monthly renewal draw. You can renew at any time. Call 091 510 2290. Good luck!

THE QL IN ITALY

Publisher Eros Forenzi reports on the active QL scene in Italy

The QL was a highly acclaimed machine in Italy when it was launched in 1984. It was supported by high street stores and big retailers up to 1987 and did well. Unfortunately, after the Sinclair sell out, Italian shops turned to the IBM PC. By 1987, about 20,000 QLs had been sold, mainly with Italian MGI roms. From 1987 onwards, more QL users abandoned their computer and opted for another machine. Only a few of these 20,000 were expert users, so we've not lost so much in expertise but only in buying power. Nowadays we estimate that there are less than 3000 active QL users in Italy, and 500 of them are expert and enthusiastic users. The main problem is English. Few of us really speak or read English. That's why few read *QL World* and only about 30 to 40 are Quanta members. Also, QL World is not easily found here.

Luckily, most of these 3000 users are willing to stay with the QL, at least until they can find some support in Italy. SPEM is still active in the QL area and provides a good range of QL peripherals of its own making and from other brands.

Then there is a very well organised club and some committed enthusiasts like me.

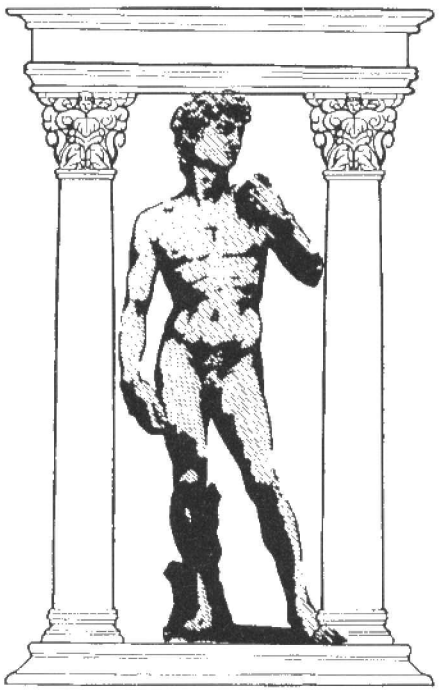
The club is Club-Ware, Chairman Mr Roberto Orlandi, with about 100 members and a good bi-monthly A5 magazine called News-Ware. Club-Ware has been active since 1987 and plans to stay with the QL for the foreseeable future. A public domain Italian QL software library has just been organised; eight disks are already available.

Then there is me. I edit and publish a public domain disk magazine called QItaly, on 3.5in or 5.25in 720K disk, on a bi-monthly basis. 14 issues are available up to now. Issue 13 was all about Qram style programs. Each issue has about 700K of stuff: articles in _doc format, PD programs, and 20 to 30 screenshots taken from commercial and PD software, to give at last some colour images of QL programs.

News-Ware magazine reaches about 300 QL users, almost all in Italy, while QItaly arrives in Germany, France, England, Belgium and even USA, and has an estimated 300-400 reader base.

In the very near future, I, Eros Forenzi, and Roberto Orlandi will join together to build up a more efficient and stable QL club. We have many innovations in mind and will let you know what happens.

We have even some small, good quality software houses. Ergon Development, for example, have produced a highly sophisticated QL disassembler that can extract any SuperBasic extension out of a commercial toolkit. The name of this product is DEA 4. Ergon have about ten good quality QL programs, and they have English prompts and manuals too! Another good program from Ergon is MasterBasic, a useful and easy to use tool for the Basic programmer.



We are just back from our second Italian QL Users' meeting, organised by Gruppo QL Grande Lago, Club Ware, SPEM and myself. The meeting was a success, apart from the rainy weather. Anyway, about 100 QL users turned up and that's good by Italian standards. We had many exhibitors: SPEM, Jochen Merz, Miracle Systems, EEC Ltd., Ergon Development, Club-Ware, QItaly Group (E Forenzi), and GQGL. We were very pleased to have Jochen Merz from Germany with his wonderful Atari ST with QL emulator, and Miracle and EEC from England with many interesting products. We also distributed some product leaflets from Jurgen Falkenberg and QLympic Computer Systems (both German firms) that could not come. Software products on display included QPAC 2, text87 Italian, QD III from J Merz, QItaly 1.3 Special Qram issue, DEA 4 and many others. We hope to organise a third meeting next year. News announced at our meeting should keep QL users happy for a long while.

Miracle is making a 'Super Trump Card' with more than 1 MB of ram, a 16 MHz 68000 processor and 16 bit ram memory. It should run 4 to 5 times faster than a QL. Price will be high, but part exchange with Trump Cards will be possible. Miracle are serious about the QL and plan even more interfaces for the future, but I cannot give any details now.

Jochen Merz has just introduced an extended QL emulator for the Atari ST, that will make it possible to have a screen resolution of 768 x 280 pixels in MODE 4. The software to drive this resolution is from Tony Tebby (Tebby means quality to me) and it will be compatible with every Qram style program. Jochen's own ST was a little beast, with 4 MB of ram, a 16 MHz 68000 processor with 8 KB of cache ram, and a 44 MB Atari Megafile hard disk with removable cartridges (reading speed about 200 KB/s). Again, the hard disk driver is from Mr Tebby, who is constantly improving the pointer environment and writing device drivers to exploit the Atari ST hardware potential. Again, Tebby and Merz have many secret places for the future, but I cannot disclose them.

So, to sum up, there are still many enthusiasts in Italy who want to stay with the QL and upgrade with the new Miracle card or Merz ST. We hope that every other QL user anywhere in the world can resist the temptation to switch to another so-called next generation machine. We have it right now, it's the Qdos based QL!

ADDRESSES

QItaly Group, Eros Forenzi, Via Valeriana 44, 23010 Berbenno (SO). Tel: (0342) 492323 (local)

Club-Ware, Roberto Orlandi, Via Brescia 26; 25039 Travagliato (BS). Tel. (030) 6863311 (local)

Ergon Development, Davide Santachiarra, Via Emilgio de Marchi 2; 42100 Reggio Emilia. Tel. (0522) 70409 (local)

Gruppo QL Grande Lago, Giovanni Zane, Viale Marco Enrico Bossi 39, 25087 Salo' (BS). Tel. (0365) 40102. Fax. (0365) 520184

SPEM, Guido Masoero, Via Aosta 86, 10154 Torino. Tel. (011) 857924 (local). Fax. (011) 280009.

THE QL IN BRUSSELS

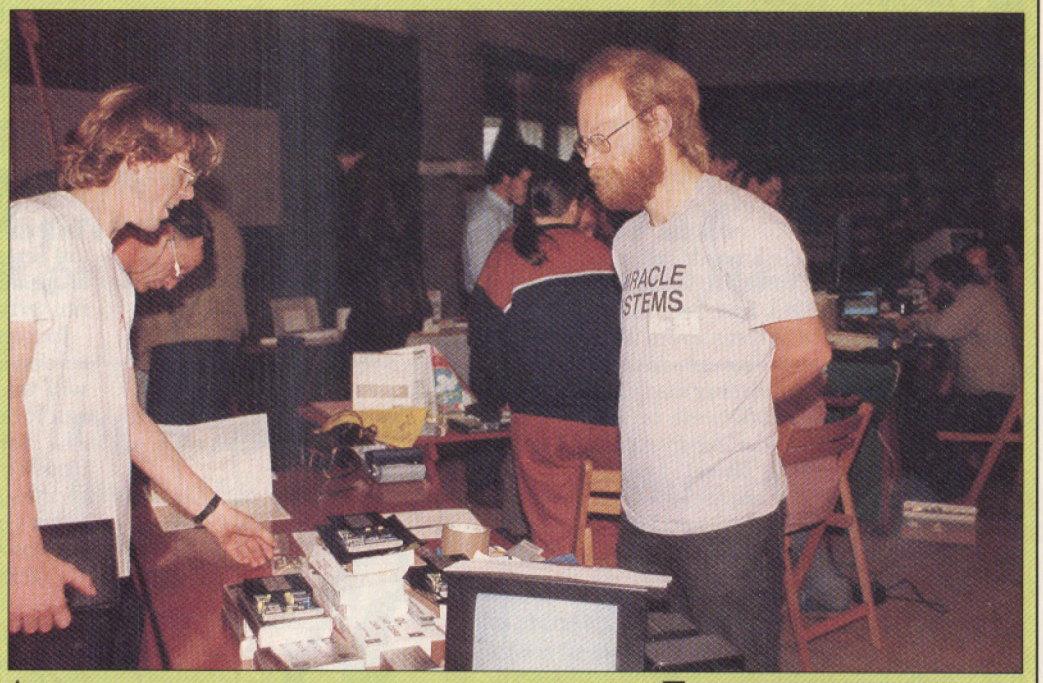
Saturday 6th October was the date for the second European Microfair, held in Vilvoorde, near Brussels in Belgium. Virtually all of the companies and user groups dealing with the QL were at the Fair, subtitled The Big Sinclair Show. Approximately 400 people came from all over Europe.

Carl Watson at the Euro Microfair

Many QL user groups made an appearance at the show, including Quanta, the British QL user group. QL Contact France brought along some copies of their magazine, *Informa*, which looked interesting, and well presented, although it is written in French of course. One of their members was demonstrating a prototype 256KB static battery – backed ram board for the QL, which allows users to keep data, resident extensions or SuperBasic extensions, for example, in memory even when the machine is switched off. This could prove useful for eprom development work, where a user often needs to blow and erase many eproms, which takes time and energy, especially if the stored data needs to be changed regularly. Sin_QL_AIR, the national Dutch QL user group, were represented and sold copies of *Quasar*, their informative newsletter. Some members of the Italian user group also came.

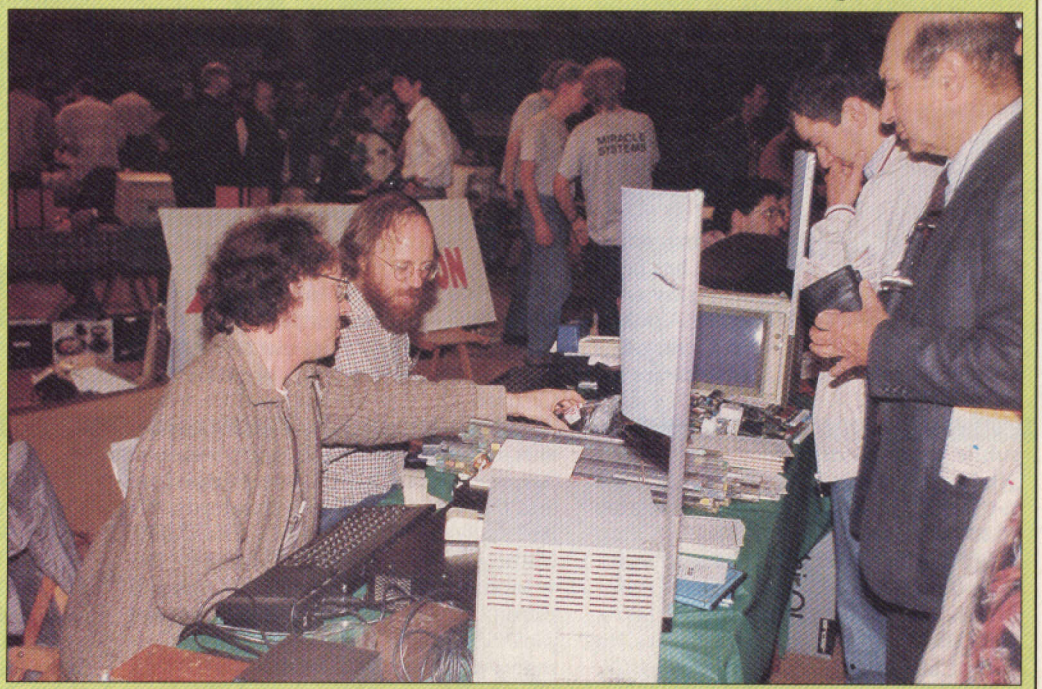
Numerous software houses made a showing. Freddy Vaccha, of Digital Precision, spent much of the day demonstrating his company's products, including Turbo, the SuperBasic compiler, and Font Enlarger, reviewed in August's *QL World*. Progs, writers and suppliers of *The Painter* package, brought their new fast database package, *DATADesign*, which runs under the Pointer Environment. DATADesign includes such features as variable fieldlength records and various types of sort, and should prove to be quite an improvement on *Archive*. A SuperBasic program is also included to convert Archive data files, although the program itself is 100% machine code and keeps all of the database resident in memory, which means it should be very fast indeed.

Jochen Merz spent the day demonstrating his and Tony Tebby's (QJump's) software, which seemed to be selling well, on a Mega ST fitted with *HyperCache*, using the QL Emulator. HyperCache is a speed up device for the Atari ST and can be fitted to most STs including those running QDOS on the emulator. The speed improvement over a



▲ Carl Watson talking to Miracle's Stuart Honeyball.

▼ Tony Firshman of TF services demonstrating QualSoft.



standard QL, as direct a result of HyperCache and the ST, was quite remarkable, as was the ability to keep all of the programs on one removable 20MB hard disk. *QPac II*, QJump's replacement for *QRam* reviewed in August's Sinclair QL World, seemed to be creating a lot of interest amongst the crowd of people that continually surrounded Merz's stand and there were whispers of an upgrade to *QD II*, Jochen Merz's excellent editor program.

Lawrence Reeves, of QView, made an appearance and sold many Minervae throughout the day, clearly demonstrating that this derivative of QDOS, which has now reached version 1.84 is rapidly becoming a must for all serious users. Its ability to multitask many copies of the SuperBasic interpreter at the same time was demonstrated, and seemed to impress all the visitors to his stand. Next to QView was Tony Firshman Services, demonstrating the *QualSoft* Terminal Software package, using a QL interfaced via the RS-232 port, to Psion Organiser, and sending data both ways.

Jochen Merz: QL emulator on the ST.

Many hardware companies were at the fair. Miracle Systems, who cycled from York to this event, sold both of the hard disks they carried with them and many Trump Cards. They also spent much time answering technical questions about the QL hardware in general. Unfortunately they didn't bring along the 68020 plug in card, for the QL, which they are said to be working on. If it appears it could result in an increase of the QL's maximum processing speed by a factor of approximately five or more. The people from TF Services spent many hours slaving over a hot soldering iron and as a result a few more people now have mended QLs. EEC Ltd were present and sold QLs, software packages and printers, proving that new interest in the QL is far from dead.

One enterprising company, Caplo Plot Applications, were using a Thor XVI, to drive a 'flat-bed' pen plotter, which spent much of its time drawing the American Space Shuttle with great accuracy and in four different colours. Nearby, a QL user was demonstrating a system he had developed which generated pseudo-three-dimensional pictures on the QL's screen. This is done by drawing everything on the screen twice, the second image offset a little from the first, and colouring one in red and one in green. The user wears red and green tinted glasses and the picture, seen in monochrome, appears to have depth. It certainly seems to be an idea with great potential, which could find much use in games and cad-type applications.

Discussions

The small cafe, in the Eurovolley Centre, was open all day, selling coffee and biscuits. It also served as a meeting place for QL enthusiasts, and discussions in many languages were to be heard late into the evening. Overall the European Microfair was well worth a visit and the organisers, Club Sinclair BruQsL of Brussels, are to be congratulated. I look forward to the one next year.

TF Services

ASTRACOM Modem

Intelligent buffered modem with text status messages. Hayes protocol, parallel printer port (can be used as 6k serial to parallel printer buffer). 240vac or 9-12vdc. Autodial/answer and many programmable registers incl printer logging of RX and/or TX data. BT approved. Can be used with any computer.
AC1000 (V21/23 v1.5a) £175
AC1000T (V21/23/22 v2.03 + tone dlng)... £257
AC1000 (V21/23/22/22bis V2.03) (tone) .. £299
V22/V22bis upgrades ..... ring for prices

QL REPAIRS

QLs tested with Thorn EMI test rig and ROM software
(MDV hardware extra if required)
£25 - 6 month guarantee

QL SPARES

Faulty QL boards (excl plug in ICs)..... £9
Keyboard membrane... £7 Circuit diags... £2
68008 cpu..... £8 1377 PAL..... £3
JM rom set..... £12 Power supply... £12
8302 ULA..... £10 8301 ULA..... £10
8049..... £7.50 MDV ULA £12
Other components (sockets etc)... plse PHONE

QuaLsoft Terminal

Viewdata/VT52/VT100 QL TERMINAL EMULATOR
'Een deluxe communicatieprogramma van Kualsoft'
Multitasking for electronic mail, PRESTEL etc, CET download. Phone directories for ALL (yes ALL) modems for the QL, autodial (where poss) and logon, two-way FILE TRANSFER to ATARI ST, IBM PC, Psion Organiser (via comms link), XMODEM, real time clock/timer, buffered logs to file/printer transmit files, editable command line, Swedish/Norwgn/German options, translates, editor etc.
SOLVES ALL PACKAGED MODEM SOFTWARE PROBLEMS
Unbuffered modems usable with Miracle Modaptor
Software (3.5" or mdv) & manual £30

BULLETIN BOARD

QBBS - The first UK QL scrolling bulletin board. Now even larger with international ECHOmail and 10mb of files (including the UHOLE King James' bible!)
071-706 2379 V21/23/22/22bis 24hrs

COMPUTER CLEANER

Much improved mains filters. 40-80 db cut & 3 spike filters. 390 joule cut (3 & 4 way)
Certified X/Y 240ac capacitors
2-way (5a adaptor)..... £14
3-way (5a adaptor)..... £18
4-way (13a trailing socket)..... £24

FILE TRANSFER

Programs to transfer text AND programs error free (XMODEM) between computers
Available for IBM PC and compatibles, ATARI ST and Sinclair QL. Connects to Psion organiser via Psion comms link
QuaLsoft program (per m/c)..... £7.50
Serial lead (name 2 computers)... £10
Complete package for 2 computers. £25

All prices include VAT and postage in the UK
MAIL ORDER ONLY

VISA
12 Bouverie Place, London W2 1RB (tel: 071-724 9053)
SCROLLING BBS FOR ORDERING/QL HELP - 071-706 2379 (All speeds to V22bis)
Fax: 071-706 2379 Telex: 265451 (ref: 72:MAG90645) Prestel: 017249053

sinclair QL

EEC LTD THE QL "ONE STOP SHOP" – PICK & MIX OFFER

QLs COMPLETE FULLY TESTED AND WITH 6 MONTHS WARRANTY – TV lead, QL Software 2.35, Quill – Word Processr, Abacus – Spread Sheet, Archive – Records, Easel – Business Graphics £125
Customers buying one of the QL Computers are given one year's free membership to QUANTA

PRINTER MT81 – Dot matrix, or C. ITOH daisywheel (with install program) £130

MONITOR – Philips 14" professional EGA high res. QL and P/C compatible with lead £220

DISK DRIVE – 3½ 1 mbyte with QL lead to trump card CR disk card, with power supply, switch and mains plug – external DIP switches and handbook £85

TWIN DRIVE – 2 of above £160

UNCASED DRIVES NEC 3½" 1Mbyte £35

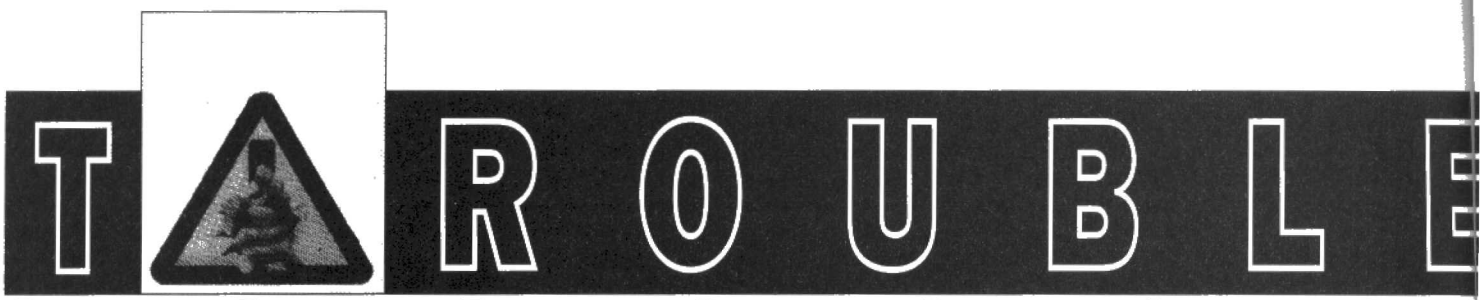
TANDATA MODEMS Q Call, Mod, Connect £40

PICK AND MIX OFFER

BUY £250 OR MORE AND HELP YOURSELF TO £20 DISCOUNT
PRICES INCLUDE VAT – SEND SAE
FOR FULLER DETAILS OF QL PRODUCT, SOFTWARE & SPARES

Prices include VAT. Terms CWO, Acess or Visa. Minimum order £10.
Carriage £8.00 for printers & QL (overseas £20.00) other items £3.00 (overseas £6.00)

EEC LTD
18-21 MISBOURNE HOUSE, CHILTERN HILL, CHALFONT ST PETER, BUCKS, SL9 9UE
Tel: 0753 888866 Fax: 0753 887149



Bryan Davies reviews a suggestion for a new specification – how well would it serve the QL?

The micro market is generally in a state of evolution rather than revolution. This seems to be one reason why Clive Sinclair is not too interested at present, he being the innovative type. The QL is around seven years old; the original PC is older. Maybe we are due for some wonderful new form of computer to arrive? The 'Next' computer was heralded as just that, but now seems to be viewed as not so great. The Apple II and Macintosh made the micro much more usable than existing computers, and the Next has some very good improvements over the Mac standard (they sprang from the same source), but it is said to lack one thing that users need – good applications programs, readily available and at a reasonable price. It is almost traditional in the QL world to denigrate both the PC and its MS-DOS operating system, but the criticism generally seems to be made by people with limited experience of either; further, they miss the point about what makes a computer attractive to buyers. The right add-on hardware and software must be available. IBM produced the PC, but it was the software writers and the clone makers that made it a huge success.

When we consider what a 'new QL' ought to have, let's not ignore the good things that about 30,000,000 PC users already have. Forget the superior attitude that says the QL is automatically better than the PC and try to be a bit objective and learn from other users. Wouldn't it be good to be able to plug any one of half a dozen types of hard or floppy disk drive into your QL? A choice of memory boards would not come amiss, either. The thing is that they would all have to conform to the *same* standards, set by the designers of the computer rather than by every designer of an add-on device.

Reader **John Berryman** has offered his suggestions for a 'QL2'. He feels any new machine ought to "sell in Dixons in vast numbers", and it is presumably true to say that a large proportion of UK QL owners did buy their QLs in either Dixons or W H Smith. Such chains have tremendous marketing clout, and they undoubtedly reach the average member of the buying public. One problem with them is that they are very profit-conscious, and they soon drop products which are not moving fast enough to meet their projections.

The early failure of Sinclair to get the QL working satisfactorily resulted in a very high percentage of returned units, and chain shops lose far too much money handling returns to stick with any product which isn't quickly made saleable. It was very noticeable when W H Smith suddenly dropped the QL; Dixons continued to sell it at discounted prices, but only with the intention of clearing out stock, not to maintain it as a line. It is unlikely any of the people currently interested in developing an updated QL would have the backing to produce machines in the quantities associated with selling through chain shops, and would almost certainly have to rely on mail order sales.

John goes on to suggest that a new QL should be housed in a box strong enough to support the display unit. The operating system firmware ought to "recognise both MS-DOS, QDOS and (say) Tebby Toolkit commands . . . which . . . should have been included in the first place".

His next idea might have designers up in arms: he wants the computer to recognise the format of an inserted disk, and write to it in its own format. That is, it should be able to accept, say, QDOS and MS-DOS disks without user intervention. The video output should enable either normal mono or very high definition displays to be connected. Options for combinations of hard disk and floppy drives should be available, with the hard disk capacity being of the order of 80 MB. Additional storage devices should be offered, such as tape streamer and laser devices (eg cd-rom). There should be external ports for mouse, dmp and daisywheel printers, and eprom devices for control of remote equipment (eg lights, machinery); a working network is a feature that few current QL users would argue with.

Expansion slots

There should be internal rom sockets for third party software. Memory should be at least 1 MB of ram, with slots for expansion memory boards. A mains filter should be built in; again, many QL users would echo the need for this. The monitor power would come from a socket on the computer. The keyboard keys should be programmable, and there should be a separate numerics pad. An internal back-up battery supply would be needed, to retain information such as password, time/date, and key assignments. Nothing contentious about that, but the idea that this battery supply should have enough power to close any open files if there were a failure of the external mains supply takes us into the area of portable/laptop computer design, where prices are much higher than normal table-top computers. Non-technical users would welcome a 'child's play' start-up menu as soon as the computer is powered up. Last on John's list was the Psion *XChange* suite. This might not meet with universal approval!

Price skirted

The matter of price was carefully skirted round, but it was suggested that the manufacturers would need to have sufficient faith to wait up to five years to recover development costs. If such a computer were to be manufactured in the UK, you could be quite certain that virtually no manufacturer would plan to wait anywhere near that length of time to recover development costs. Well, what do you think about John's suggestions? One thing he didn't acknowledge is that, despite his criticisms of IBM-type PCs, some of his features are basic to even the cheapest PC, and such systems cost less than £500 complete with display. In addition, the Amstrad PCW is still selling quite well at around this price, complete with printer. If we were to take guidance from successful computer system concepts, in the sense of their being sold in millions of units, these would be the two to look at.

The Files part of the *TaskMaster* program has been upgraded to deal with hard disk sub-directories more satisfactorily. It should not be necessary in future for the program to format an entire hard disk when making a working copy to it, either! This is **not** an upgrade in the sense that the current version of TaskMaster is being offered at a special price to existing users, as most of them would not benefit from the changes, but – as hard disks get more common on QLs – the program will be more helpful to those users lucky enough to have obtained this ad-on. When placing an order for TaskMaster, you need to specify whether or not it is to be used with hard disk; if it is, the hard disk version of the program is required, and that costs

SHOOTER

M S O L V E D

£40 instead of £25. Users of the QL emulator on the Atari ST may have a problem running TaskMaster, and this is not solved by buying the latest version of TaskMaster, but can be easily fixed by a small change to the Boot routine (deleting a CALL statement; see the Quanta magazine of October 1990). The problem relates to the lack of microdrives on the ST! It apparently occurs only with late versions of the emulator, having additional device drivers.

Digital Precision are now offering a toolbox package to go with *Professional Publisher*. For £30 you get a large selection of both low- and high-definition founts, additional clip art, and various utility routines. One routine allows the user to tailor text from other sources (eg *The Editor)* to suit Professional Publisher, before loading it into that program. Another permits clip art from other sources to be utilised. Working in the reverse direction, there is a routine to extract screens from Professional Publisher so that they may be used in programs such as DP's own *Eye-Q.* The whole dtp package of *Professional Publisher, Font Enlarger, Grafix* and the Toolbox is available for £119.95. Expect the announcement of another new program from this supplier around now (see *QL Scene*) it will fill what has been arguably the most obvious gap in DP's product line.

Like Quill

Taking a closer look at *ArchRtm,* the runtime version of *Archive*: while writing about Procedures, I was reminded that there is a difference from Archive in the behaviour when the program is started. ArchRtm is rather like *Quill,* in that it can corner all the available ram memory space, so that a request in a Procedure to Backup a file to ram meets with no success. This occurs when the ram disk routine in the system is of the *dynamic* type, and forces one to use the Format command beforehand, to reserve space for any database that may have to be copied into ram. An unfortunate effect of this is that the ability of the dynamic ram disk to cope with increasing database size is lost. This problem does not occur when Archive or *ArchDev* are used. ArchRtm can be given the 'EX' treatment, like the others; that is, you can convert it to being usable with EXEC as well as EXEC_W, so that it can be multitasked (see *DIY Toolkit,* October 1989).

Version 2 of *Fleet Tactical Command* should now be available. It is usable on the QL emulators for the Atari ST and the Amiga, as well as on the QL. Some projected future developments are a German language version, 'native' versions for the ST and Amiga, and a player-versus-computer facility. The German dealer for FTC is QLympic.

Continuing the comments on laser printing, I now have first-hand experience and am generally very pleased, but the delivery of my Epson GQ-5000 was a saga of uncomfortable proportions. It involved the carrier Parceline, which succeeded in losing the first printer, but did not report it to the suppliers. The next step was to fail to deliver the replacement (one week late) 'first thing Monday morning'; in fact, the delivery driver apparently reported that the street in which I have lived for 20 years 'does not exist'. This was followed by a further attempt in the afternoon of the same day, when the driver 'had a windscreen shatter'. Need I add that neither supplier nor carrier had called me to ask for directions, or to advise on the situation, throughout this period? The final act was delivery by taxi, successfully completed in the evening of the same day. The taxi driver was stated to have left the depot at 1530 hours – he informed me that he didn't even get there till 1650. Rest assured that dealing with complaints from readers about suppliers does not give one any privileges whatsoever when it comes to buying goods oneself!

Back to the printer itself; it is considerably more complex than a dot-matrix one, and takes up more room, but the setting-up process went smoothly and the test prints were all excellent. The results really are in a different league from what I had become used to with the dmp printer. As expected, prints of graphics take a long time to appear, but are worth the wait.

One very handy feature of the GQ-5000 is that it has an emulation of the Epson FX-800 dmp (similar to the FX-80 in codes), permitting printer-drivers for the dmp to be used with the laser. That is, existing documents can be printed straight off, with no need to change the driver or reformat the document. So far, my comments have been in connection with using the GQ with a PC/AT and its programs, and it would be fair to say that I've had less problems getting printout to look the way I want it than when starting to use many new programs with the dot-matrix in the past. One reason for this is that Epson accepted the fact that software suppliers had not been creating printer-driver routines for the GQ, and decided to write the drivers themselves, so that I was provided with drivers on disk for three major programs.

Turning to the QL, I did not at first try to print from it, reasoning that I'd need the GQ-5000 driver for *text*[87] to be able to get sensible output from it. However, while writing a program review I decided to see how a screen dump turned out on the GQ, and found that it was much the same as on the dmp; it was rather more readable, but that might have been because the paper was better. Progressing to printing a Quill updates file for the program being reviewed, I loaded that into text[87] and sent it to the printer as it stood – that is, without doing anything whatsoever to the format of the document on the screen, and just setting the printer to FX Mode. To my surprise, the result was fine, coming out in proportionally-spaced 10-pitch, as was shown by the program's on-screen typestyle indicator. Changing various parts of the text to different typestyles produced equally good results. Every typestyle came out as might be expected from the screen indications. This seemed too good to be true, so I printed the same document direct from Quill. It came out fine. Another document printed direct from The Editor (not using the separate printer driver) also came out fine; as the print codes were still in the file, and there was no driver to interpret them for the printer, it was not surprising that the typestyle used changed from 10-pitch to 16.66-pitch at one point.

Laser success

I am still amazed at how little hassle the laser printer has given me. It looks as though one doesn't need to wait for special drivers to be written, as existing FX80-compatible drivers work well. The FX mode does not utilise some good features of the printer, however; you cannot 'scale' founts, for instance, so I couldn't print characters three inches high (this doesn't exactly upset me). Of more concern is the fact that the scalable proportionally-spaced Times Roman and Helvetica founts can't be used; I intend to use the latter whenever possible because it has no serifs and is very readable even down to about 5/72in high. Maybe the text[87] GQ-5000 driver will enable these features to be used. I hope to be able to report on this driver, and on printing graphics, in the next issue.

It would be helpful to receive some comment from readers who have used a hard disk with a QL. My enthusiasm for

writing some helpful tips on setting a hard disk up was considerably dampened after a few days battle, and there is no sign of a breakthrough yet. Could it be that one can, sensibly, only tackle the project on the basis of using QRam/QPac? Hopefully, by the time you read this, I'll have found another few days spare to check that thought out. Meanwhile, I continue to boot from a floppy and ignore the sub-directories that have been set up.

Di-Ren, suppliers of the *Fleet Tactical Command* warship game, have demonstrated their current version of the program to me, and it has quite a bunch of significant improvements over the original. It looks generally snappier, and more useful information is supplied for following the progress of a battle. I was informed that my particular JS QL was encouraging much more distracting activity than was usual on the authors' machines. That is, there seem to be many more damage reports, crew mutinies, and so on. Do we now have to contend with individual personalities in our QLs as well?!

The same supplier provided a new piece of hardware for evaluation. This is called a Micro-Computer Process Controller, and it is basically a set of six mains-voltage switches, which can be controlled from any computer. The Controller is supplied with one male and one female Centronics (parallel) connector, enabling it to be connected in series with any printer which is connected to a parallel port. This includes a printer connected to the QL serial port through a Miracle (or other type) serial-parallel interface, which is how most printers are connected to QLs.

Process controller

The Controller is powered either by two 9-volt PP3 batteries fitted internally, or by an equivalent external 9V power supply. The idea is that you use a simple program in the computer to send signals down the Centronics cable to operate the switches at selected times. A suitable program is available for the QL. There is a limit of three amps on the load which can be carried through the six power sockets in the Controller; the sockets are 2-pin 'figure-of-eight' type, as used with some dc power supplies. The switched voltage could be less than the mains 240V, but the load limit must be observed.

As to what you would switch, and when, that is for the user to think about. Di-Ren suggest security switching about the home, (lights or radio, for example), time-lapse photography, and even control of domestic appliances via a Psion Organiser and modem. 'Critical timing', to 1/50th of a second, is possible. A switch on the Controller allows the switch-control activity to be de-activated so that the printer can be used as normal; if the printer is used with this switch set for the Controller, there is a limitation on the character codes which can be sent down the Centronics cable, and it is that character codes above 127 will be ignored (normal characters are all below code 127, but European characters with accents, for instance, have higher codes). Both the new version of FTC and the Process Controller should be available around Christmas time.

INFORMATION:
Fleet Tactical Command (check Di-Ren for prices of versions 1 and 2);
Micro-Computer Process Controller £49.95, Controller Utility program £9.95:
Di-Ren
43 Davids Road
Forest Hill
London SE23 3EP.
Tel (081)-291-3751

Readers' Letters

The **SUB** saga continues. **MPL Wall** wrote to say that the local Consumer Protection Department had contacted one of the principals of that supplier and been told that Mr Wall's QL was available for collection, although the requested repair had not been carried out. Unfortunately, the person who discovered this apparently did not ask to take the QL away there and then, when offered it, and has been unable to make contact again. Although it was confirmed that SUB is no longer active, a bank having forced the company to cease trading, there was some doubt about whether or not the company is in liquidation; a decision on that was said to be due in early October 1990.

C Garnett comments that his name has not been mentioned in this column as a sufferer from the **PDQL** problem. He was another reader to order Turbo-Plus Quill, and get nothing in return. **TK Computerware** now handle this program. **A C Kwant** from Luxembourg adds his name to the list of those who failed to get ordered goods from PDQL; in his case, it was the Archive Tutor, and the ArchDev/ArchRtm programs. As is frequent in such letters, he comments that his cheque was cashed promptly. **Lufti Okan** from Istanbul states that he ordered SuperBasic C-Port and the C compiler from PDQL and had the cost deducted from his VISA account, and an attempt was made to deduct the same amount again (fortunately, this was blocked). Needless to say, he has not received the ordered programs. **John Hadley** is more cheerful; after waiting "many weeks and many telephone calls", he got the program he ordered (Home Budget) from PDQL, and he feels this was not entirely unconnected with his threat to make a claim against his credit card company. The author of Home Budget ended his relationship with PDQL at least three months ago. Users who are interested in the latest version of Home Budget should contact **Dilwyn Jones Computing**; DJC (see their recent adverts) have also taken over various other titles which used to be handled by PDQL.

Hadley responded to a query by **J K Easlea** about the Home Finance program from Buzzz Software. He has obtained a version for an expanded QL from the author, **Francis Ainley.** We have been given this address for **Buzzz Software: P.O. Box 139, Oxford OX2 8LD (Tel. (0865)-56083).**

There have been some complaints about service from **Sector Software** recently, but they are being dealt with (not as quickly as some customers would like, however). **David Youngquist** from Iowa in the USA, wrote twice in October, the first time to say he still hadn't received FlashBack 2 a couple of months after I contacted Sector Software about his complaint, and the second time to say he had just received the program and is most impressed with it. He is contemplating placing another order with Sector now. I have been assured that current customer complaints will be dealt with; the problem has been the familiar one of too many jobs being done by one person, and extra hands have apparently been taken on to improve the situation.

Minerva rumbles

There are said to be several hundred Minerva roms in use, and the Quanta magazine regularly carries enthusiastic letters about them, yet I get virtually no correspondence about this rom at all. This is perhaps because QL World readers are mostly not (serious) programmers and are not in the market for programmers' tools. In one sense, no news is good news, but there have been some rumblings from software suppliers that incompatibilities are occurring with existing programs (*SuperCharg*ed programs, *Touch Typist, SpellBound 2,* and *Microbridge* have been mentioned). *(Also Turbo and Fleet Tactical Command – Ed.).* The suggestion is that some of the (numerous) versions of Minerva effectively alter the ground rules of QL operation, which would create problems running some programs; the response to that could be that some programs 'break QDOS rules' but, even were that the case, it is hardly a good thing to make firmware changes now which affect programs written years ago. Comments from users would be appreciated.

Gerard Delaney has a printing problem with his combination of text^{87} V3.01 and a Kaga-Taxan printer. The problem occurs with both the NLQ and draft PW1080 drivers supplied with the program. I have checked his copies of program and drivers, in case there was some file corruption in them, and the program wouldn't run at all on my system. At the time of writing, I am awaiting his comments on running copies of mine. My own experience suggests intermittent bad connections somewhere en route from QL to Kaga as being the cause of the print errors. It would be helpful to hear from any other readers with this combination of software and hardware and having experience of printing problems.

PSION SOLUTIONS

Howard Clase has an answer for Quill and Abacus.

Living as I do in a small coastal town on the wrong side of the Atlantic with my nearest QL-owning neighbours at least five hundred miles away in opposite directions I was somewhat surprised when QL World's editor asked if I would like to contribute to *Psion Solutions*. My isolation has left me largely to my own devices and I have had to work things out for myself, with the help of *QL World* and Quanta of course. *Quill* and *Abacus* are my particular work-horses both at work (teaching chemistry) and home. I find Quill better able to cope with the complexities of chemical formulae than most of the word processors my PC-clone owning colleagues use, and Abacus has been invaluable in keeping my class records in order, right down to lists of final grades. So I have never needed to look at the alternatives.

I have learned to get the most out of the Psion programs just as they are. I am happy operating at the nut and bolts level, and not too bothered about the chrome plating.

Nevertheless there are three extras that I have found invaluable: extra memory, a disk drive and Tony Tebby's *SuperToolkit II*, especially its ALTKEY command, and I would strongly recommend them to anyone wanting to put the Psion Suite to serious use. It is also useful, although not essential, to be able to multi-task the programs using *Taskmaster* or something similar - I use a simple but effective routine that Tony Lambord put into the Quanta library that enables me to hop in and out of Quill at will.

Since there hasn't been time for me to receive any queries from the postbag, this column is based on items from recent issues of QL World.

The lack of an Export command in any of the versions of Quill between 2.0 and 2.35 is often the subject of queries and comment, most recently in *Troubleshooter* in the July '90 QLW. Why was it omitted from Quill when it is present in the other three members of the suite? I think that the primary reason is that Quill almost fills the memory of an unexpanded QL, and something had to be left out. The other obvious omission is one I would give even higher priority to, an Extract command that would enable part of a document to be saved in Quill format for merging into another file. This argument is supported by the fact that both are included in the full *Xchange* suite for the Thor, where more memory is available. It is possible to get round these omissions but, before I explain how, let's look into the Import-Export business a little more.

Export is an important command for *Archive*, *Abacus* and *Easel* since they are all designed for the manipulation of similar sets of data organised in a regular pattern: a number of similar records containing a predetermined number of values (see the Information section at the end of the QL manual). The values may be text, but are more importantly numbers which are recognised as such; each of the three programs is designed to manipulate these numbers in its own way, so it makes sense to be able to transfer data among them. I'll come back to this topic in a future article.

Quill is a pure text processor, and does not treat numbers any differently from words. As far as it is concerned, they are just strings of characters. This is why the other programs need to know, when Exporting, if the final destination is to be Quill, as the format of the _exp file is different. A file for importing into Quill should be a simple Ascii file consisting solely of printable characters and an eol (end of line) marker — the QL uses a single LF (line feed = CHR$(10)) for this. As well as files exported from Archive or Abacus a SuperBasic listing is in this format and can be Imported into a Quill file. This makes sense, because you might want to include a Basic listing in the middle of your letter to *Open Channel*, or add an index generated with Archive or Abacus to the end of the book you have just written.

Quill's Import routine is also clever enough to deal with the other common eol markers: CR (carriage return = CHR$(13)) alone (in BBC files) or CR and LF (in either order) in MS-DOS and CP/M files. So you can also Import Ascii files received via the RS232 (eg ser1) from other machines. When Importing in versions 2.1 and 2.35 you are given the option of Importing by line, when each line of the original actually becomes a separate paragraph in Quill, or by paragraph, when the lines are run together into paragraphs until the routine meets two eols together (a blank line in the original Ascii) This is treated as a new paragraph. I seem to remember that earlier versions may be different. If you still are using one of those you will have to experiment to see what happens.

It doesn't make sense to use Quill to type up data for the other Psion programs, as it is easier to do that directly. (I hope that no-one with access to Archive or Abacus would try to generate an index using Quill!) Although it is possible to write Basic programs using Quill there is no syntax checking, so you are likely to get a lot of MISTakes appearing when you load it, but you could use it to write C source code. In that case, or if you want to be able to transfer your Quill files via the RS232 port into a Beeb or an IBM, you will miss the Export command.

You cannot send a Quill file directly down the RS232 because it includes all sorts of information about the format of the file - headers, margins, print styles, etc, not to mention the 20 byte file header full of control codes intended for Quill's eyes only that would really give indigestion to any alien wordprocessor you fed it to. To produce a simple Ascii file you need to strip all this out, and that is just what the printer driver is able to do, given the appropriate printer_dat file. Try the command:

```
COPY quill_doc,scr_
```

on one of your shorter Quill _docs and you will see what I mean; the information is there, but so is a lot of other guff, and the formatting is non-existent. On the other hand if you try the same thing on a straight Ascii file it should come out right.

I know that many beginners throw up their hands in horror when the dreaded install_bas is mentioned, but it is worth getting to grips with if you want to make the most of your QL. I admit it isn't as user friendly as I would want it to be, but there are few word processors that allow the user the flexiblity of writing her own printer drivers. (Just imagine if you had to pay Psion £10 for each extra driver you needed, and that's peanuts compared with what some MS-DOS software companies ask!) One thing I have done is to Q_Liberate Install_bas to speed it up a bit; I don't have access to DP's compilers, so I do not know if they work on the unmodifed program, but I imagine there would be little trouble. (Perhaps someone could try, and let me know what happens?) Anyway this is just about the simplest printer driver you will ever have to write.

First, make sure that if there is already a printer_dat file on your disk it isn't important, since it will be overwritten. If you are not sure then type

```
COPY dev#_Printer_dat,
scr_gscr_
```

I shall use dev# = mdv1, flp2, ram8 etc throughout — you will have to put in the device appropriate to your system and the file will be written to the screen, much of it as gobbledegook, but the driver name will appear fairly near the beginning; if you don't want to lose it then COPY it to another file with a different name (eg EPSON_dat). Then LOAD your copy of Install_bas, and tell it all it needs to know until you hit the main menu.

Make sure that OTHER or DEFAULT is highlighted (use the up/down arrow keys if not),

and press F1(COPY) to create a new driver. Pressing the up/down arrows moves the highlight in the direction of the arrow. Left or right arrow enables you to alter the entry, when one of three things may happen: 1: the entry may toggle among a limited number of options 2: a numerical value may decrease (left) or increase (right) or 3: the entry may vanish completely ready for you to type in a new one.

Give the driver a new name. PORT, BAUD, PARITY and CONTINUOUS FORMS are immaterial unless you want to use the RS232 directly from Quill; set LINES/PAGE to 0 and CHARACTERS/LINE to 80 (if you are eventually going to be using a tv to view the file you might want to use 40 or 60). Finally you come to the END OF LINE CODE, and that depends upon what you intend to do with the file. For normal QL use use LF alone. If you itend to send it to an MS-DOS machine then use LF,CR. *Everything else* should be NONE. Press F5 to install the driver; your new driver will now appear on your disk as "printer_dat". Copy it at once to a new named file, eg ASCII_dat (or rename it if you have SuperToolkit II).

One of the problems with the Psion suite is that they all use a file called "printer_dat", and you may want to use a different one with each program, or even switch in mid operation. Ron Massey's approach was to modify install_bas and the Psion programs themselves, but you can switch in mid-stream. I prefer to give each of my drivers a characteristic name, and then move them into place when needed. You can do this with the Backup facility in the Files command that is present in all the programs. This works just like the COPY command in Qdos, except that the file names must conform to the Psion convention of no more than eight normal letters or numbers followed by a three letter extension (oddly any three printable characters are allowed here — eg _ $&!). So you press {<F3>, <F3>, <F>, <B>}, backup ASCII_dat to dev#1_printer_dat, and there you are.

Don't print your file just yet, as there is a bit more massaging to do first. We've eliminated all the special codes for special typefaces (the letters will still be in the file but they won't be bold, underlined etc.), but there are still margins, justification, headers and footers to deal with. First SAVE a copy of the document if you are ever likely to want it again in regular Quill format or you will have to reverse all that follows, and it's tedious. Set left justification to remove the extra spaces Quill inserts otherwise {<F3>,<J>, <L>, ENTER} and move both the indent (new paragraph) and left margins as far left as they will go to avoid a row of spaces at the beginning of each line {<F3>,<M>, as many left arrows as needed, SPACE BAR, left arrows again, ENTER}. You can leave the right margin where it is. Headers and footers should both be set to none {<F3>,<H> (or <M>), and SPACE BAR until "none" appears then ENTER}. Finally go into the display menu {<F3>,<D>, and set both the upper margin and bottom margin to zero then ENTER}. (I usually set page size to zero too, but you must not do this if you intend to Extract - see below.) Now you can print!

{<F3>, <P>, ENTER, ENTER}. If you want to Export (print to a file), type the filename in the form:

[dev#_]filename[_3-1etter-extension]

The bits in square brackets are optional; if you don't override them you will get the default device (where your _doc files usually end up) and the extension _lis. The file you have produced ought to be exactly what you would get if you had the Export command. If you have your IBM clone panting away waiting for input at the other end of the RS232 then just add a third ENTER, Quill cannot tell what is actually at the other end. If it thinks it's a printer there's no reason to disillusion it. (This will only work if the BAUD, PARITY etc. are correct, but if you are trying something like this you are probably experienced enough to know what you are doing.

The same procedure can be used as a poor man's Extract, although it is rather fiddly, and it might be just as easy to type in a small piece of text again. You don't have to Export all of your document, of course; since you are using the Print command you can select one or more pages. Remember to save a copy of your document before you alter its format. Then set a page break before and after the passage you want to Extract {put the cursor on the line above where you want the break and press <F3>, <F3>, <P>, <P>}, note the page numbers, and put in the appropriate values when Printing. You only need to adjust the margins and justification etc. as above for the section you want to Extract. You can then Import the text back into another Quill _doc in the usual way. Since the Exported file contains none of the formatting or type face information you will have to put all this back in by hand. This is always true when you Import; for example, the left and indent margins are set to the left edge of the screen.

Improving Abacus

Like Thomas Sutton (Open Channel Sept '90) I am a fan of Abacus, and would like to see an enhanced version - although I don't think it is very likely, since Psion have clearly lost interest in the QL. Interestingly, although there are alternatives to Quill and Archive, no other QL spreadsheet has yet appeared. One thing I would like to add is lookup tables that can be addressed absolutely rather than relatively. It wouldn't be useful for Abacus to distinguish between text and numerical values automatically as Mr Sutton suggests, because it is sometimes convenient to enter numbers as text when you want the value to remain on the screen as a reminder, but not to be included in a sum or average, as, from my experience, when dropping the lowest mark from a student's record. In the meantime, if he has Toolkit 2, one or two of his other wishes can be granted.

You can use the ALTKEY command in your boot to create macros for any of the Psion programs. This enables a string of characters to be fed into the keyboard buffer when you are holding down ALT at the same time as you press another key. Although it is set up using a Basic command the ALTKEY definitions remain, during another job, and even after NEW. The real power comes when you discover that you are not restricted to printable characters in your string, but can include the code corresponding to any key (or combination) that generates a code when it is pressed; and that is most of them except keys like TAB and SHIFT. The only problem is that you cannot insert them into your string by pressing the keys themselves, but have to use the CHR$() function.

To find out which code is generated by which key(s) type in the following command in SuperBasic:

```
REP i: PRINT CODE (INKEY $ (-1)),
```

This little instant program will print on the screen the code of any key (or combination) you press until you BREAK CTRL SPACE. (Or you can look up the code in the Concepts section of the QL manual!) For example, you will find that pressing <F3> generates 240, ENTER is 10, and down arrow is 216. These codes can then be inserted into a string as in:

```
ALTKEY 'k', CHR$(10)&CHR$ (216)
```

If you include this in your Abacus boot then pressing ALT, k will enter the value and move to the cell below. Other keys can be similarly defined to move in other directions, but unfortunately not the arrow keys themselves. I use h, j, k and l because these perform this role in a mainframe editor I sometimes use. The other command Mr Sutton asked for was one to go to the first cell in the next line. This can be provided in a similar way, what is needed is an ENTER, followed by several Left arrrows and a down arrow. Try:

```
ALTKEY 'r', CHR$ (10) & FILL$ (CHR$ (192), 10) & CHR$ (216)
```

Note the use of the FILL$() function for the multiple left arrows; this will give you 10 left arrows, which ought to be enough, but you can easily adjust the number - it doesn't matter if there are too many. You can do much fancier things using this technique: how about saving a copy of your Quill document and returning the cursor to exactly where it was from a single ALTKEY keypress? But that will have to wait.

Buying *for* Beginners

Felix Fonteyn scans the best QL hardware advisers.

The Sinclair QL was launched by Sir Clive Sinclair in January 1984 and was proclaimed as a quantum leap in the computer world. That is, it was a leap from the Spectrum era, and left behind most of the other popular computers of the time as well. In terms of simplicity, capability, and user-friendliness it is only recently that the competition has come near to matching it. The QL system included QDOS (the operating system), SuperBasic, and the Psion software, as well as a range of programs for business, leisure, and programming by several top level companies such as Sage and Hisoft. The concept was almost as innovative as that of the Mini in the motoring world. Ironically, it is partly due to Sir Clive's later financial problems with Sinclair Research Ltd. that the QL is still available and at a very competitive price today.

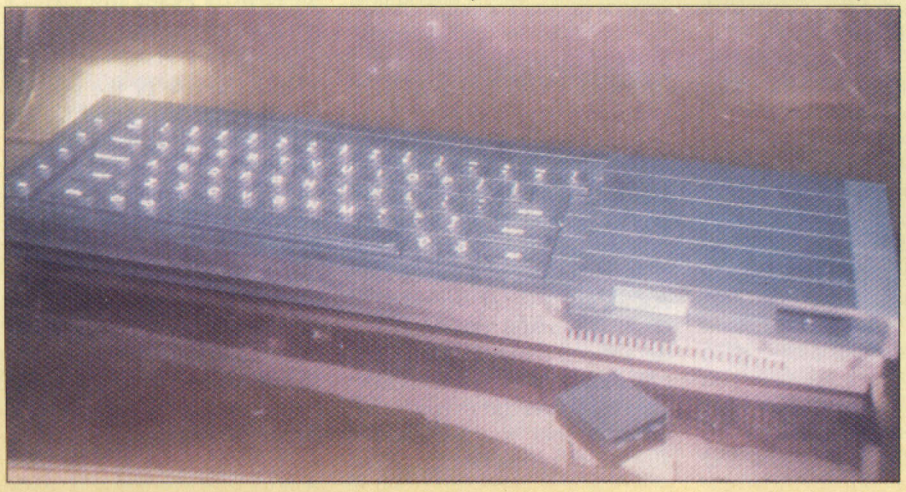
Though the leap was a genuine success, the launch was a flop. Even today it is not easy to explain why the QL did not follow the phenomenal popularity of the Spectrum range, which of course is far from extinct. Several reasons can be offered, such as Spectrum users not upgrading to QLs, the majority being mainly games players used to a choice of programs which would not play on the QL. Also, the QL was launched ahead of its time with the title 'Sinclair QL Professional Computer' which can be seen embossed on the underside of each unit. Yet various amendments were still to be made to the hardware at the time of the launch, which discouraged some buyers.

Sir Clive's intention was clear: he wanted to do for the small businessman what he had achieved with the home computer; many took advantage of the offer, and are still using QLs today. However the response was not sufficient to establish the QL as a leader, as the public at that time were not yet computer-minded enough where business was concerned. Contributory factors were production delays, and an unsympathetic press reception when it was seen that the hardware was incomplete.

The original QL package consisted of the QL computer, its power supply, the Psion Quartet software, the word processor *Quill*, spreadsheet *Abacus*, database *Archive* and graphics program *Easel*, a User Guide, and leads packed neatly in its silver and black box. It was then, and still is, possible to unpack the box, plug into the mains and a tv and be word processing, setting up a spreadsheet and database over a weekend without any special computer knowledge; just a little courage, perhaps. The following week will see a purchaser looking for a suitable printer.

The same QL packages are still available from EEC Ltd. today at the very reasonable price of £125. The QL computer itself is a compact integrated unit of keyboard, electronics, and two built-in drives using small and relatively fast Microdrive units and miniature cartridges. These were a great improvement on cassettes, being tape loops which can load and save almost as fast as disks, though they have less storage capacity and require more careful use. The memory at 128K is adequate for most ordinary purposes initially and can be expanded to 640K when required with Miracle System's *Expanderam* or to as much as 896K with the *Trump Card*, which has valuable extra features including a disk drive interface.

In its standard state the QL can operate printers, modems, and other devices. Just to prove the point, this article was written with a standard QL using Quill and microdrive cartridges. When expanded it can utilise more elaborate programs and equipment such as a mouse or scanner for copying photos or drawings to the screen.



The QL has a built-in feature which is of special interest to businesses and schools: the ability to communicate with each other through a network of up to 64 stations on simple twin cable to a distance of 20 metres or more between stations. Another feature, and one which makes the QL particularly attractive to many users is that it can multitask: that is, run two or more programs at the same time. For example, it is possible to have a word processor running at the same time as a spread sheet and work on both. The versatile design of the QL enables it to accept peripheral equipment of almost any kind; disk drives, PC-style keyboard, even a hard disk drive can be added.

There are many high grade, reasonably priced programs for all purposes from many sources. For example, some twenty or more including a PC Emulator are supplied by Digital Precision. New programs are still appearing in *Sinclair QL World* magazine; there is even a new source, Dylwyn Jones Computing. GCH Services specialise in games. The advertisements in QL World are the place to look for the established proprietary software; there are about twenty sources. It is also the magazine used by the suppliers of accessories and hardware.

From the start the users of the QL recognised its features and possibilities for development and were experimenting and producing all kinds of additional hardware and software. Back in 1984 a group of users got together with some of the QLs designers and formed IQLUG (The Independent QL User Group) which originated the Quanta monthly Newsletter. QUANTA now has 2000 members world wide. There are twenty local subgroups in the UK and several on the continent. Equipment and program suppliers regularly attend to demonstrate their products. The QL cult has spread around the world; EEC alone reports dealings with independent groups in the USA, South Africa, Australia, and New Zealand.

As is now well known, Amstrad bought the Sinclair computer name and business from Sinclair Research Ltd., which in itself was an oblique tribute to Sir Clive's pioneering. Amstrad continued to market and develop the Spectrum as a games machine, but dropped the QL. This would normally signal the demise of a computer, but not in this case. It is still in demand, though not in sufficient quantities to justify it being available in the high street shops. The main supplier of brand new machines is now EEC Ltd. by mail order; they have sold 500 machines in the last 18 months.

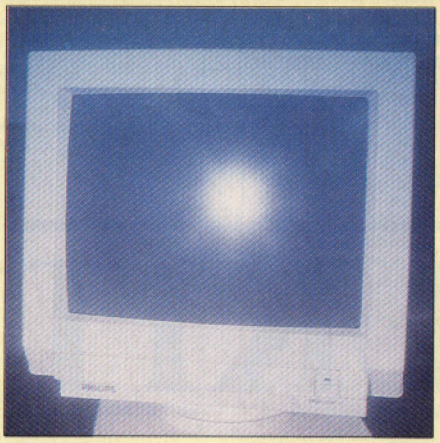
When production of QLs stopped the stocks were sold off with a considerable amount of spares and accessories. The demand for QLs justified the recovery of

the stocks, and production of undated and tested QLs.

One of EEC's unexpected successes in this area was a QL assembled from available components and advertised as a "Backup QL". Many confirmed QL users jumped at the opportunity to back up their system with a spare, to network, or have two QLs doing different jobs rather than invest a considerable sum in a new system. Even today the less well-off computer users who took the trouble to enquire about and acquire a QL some years ago can search for an alternative, and find it hard to see any advantage in 'upgrading' to a different system. There is still sufficient stock of new QLs to assure the supply for some time yet at the present rate of sales.

Scares have appeared from time to time concerning the supply of Microdrive cartridges, with which the QL functions in its basic version. If they became unobtainable at this moment it would make no difference to the progress of the QL, as nowadays 3.5in disk systems which can be plugged in are relatively inexpensive, and in many ways preferable. But there are an estimated 20,000 new cartridges obtainable at this time from companies such as Sector Software and EEC, which was instrumental in ensuring that production continued.

The ambition of regular users of the QL will be to change to 3.5in disks now that they are so cheap, reliable, and have good storage capacity. There is a choice of several disk drive units. The cheapest combination of interface and drive at the moment is probably EEC's single drive



A Philips 9CM073 EGA monitor

with the Miracle Systems QL Disk Card at £190, which includes *Toolkit 2* that adds a hundred or more useful enhancements for operating the QL. Extra drives can be added if and when required. The most popular system is the top of the range Trump Card 768K package by Miracle Systems at £375 which, besides giving extra memory and Toolkit 2, has a ram disk which makes it possible to greatly speed up work, especially with files.

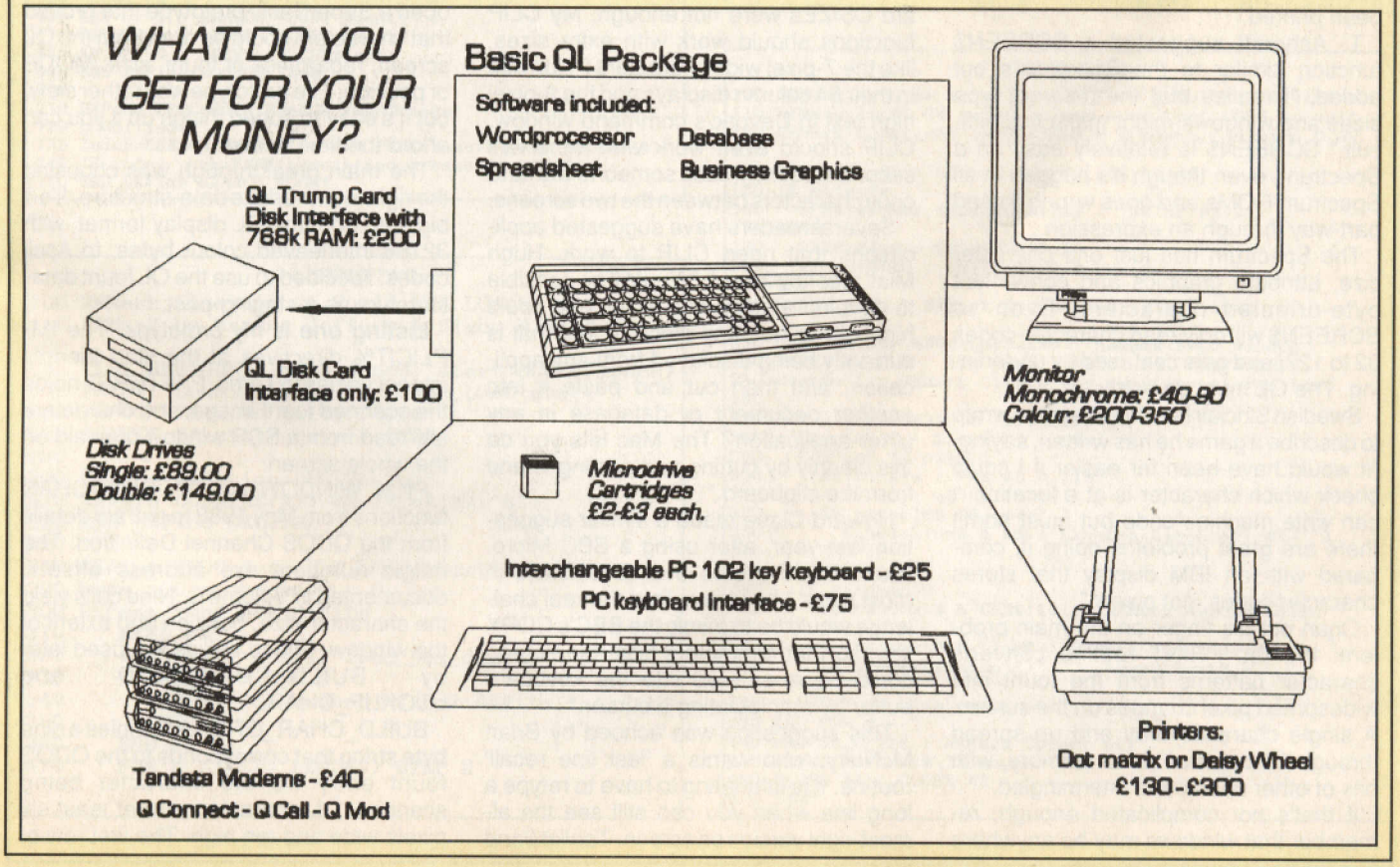
There are many types of peripheral available for the QL from regular QL dealers and some other sources. The mono BM 7502 (green screen) and BM 7522 (amber screen) are stocked by Care Electronics currently at £97.75 each. Popular Philips colour monitors are the CM 8833 14 inch medium res RGB composite, stocked by EEC Ltd. at £240, and the PRO 9CM073 (see picture) 14 inch high res CGA/EGA 0.31 pitch colour monitor at £220 with QL lead. The latter is good value at £50 below the official list price, but does not have stereo sound.

Star printers have found favour with QL users. The Star LC10 mono, Star LC10 colour, and Star LC 2410 24-pin printers are stocked by Care and Sector (around £180, £225 and £300 respectively), and EEC stock the Mannesman Tally MT81 130 cps (26 cps NLQ) autoload, fanfold and cut sheet loading printer, daisywheel or dot matrix, for £130 with a QL installation program and QL lead.

TK Computerware will also recommend upgrade hardware although they do not regularly advertise hardware selections. TK expects to be carrying Jurgen Falkenberg's new QL-Keyboard-90 interface for connecting normal XT- and AT- keyboards to the QL.

Tony Fishman Services carries a range of mains filters and spike suppressors for the QL from £14 to £24, and the Astracom QL modem from £175 to £299.

There was a period of despondency in the QL's career when Amstrad took over. Users feared that the end was in sight. However, time has shown that Sir Clive's concept of a practical system was right, as the QL has gained strength ever since. The threat of extinction brought together the main forces behind the independent QL, who continue to serve the QL user. Measured in terms of new Quanta members and recent sales of hardware, the QL user's position is as good if not better than it has ever been, and exists in a friendly club-like atmosphere.



DIY TOOLKIT

Simon Goodwin introduces Super-Basic functions that ease data transfer for all QDOS programs that use the screen and keyboard.

After years of experiment, I've fashioned another missing link for QL and Thor users – fast Basic functions to read characters from any window. In this article I shall explain how the functions were developed, and how they can be used. Next issue I will explore the assembly code and further implications.

These functions have often been suggested by *DIY Toolkit* readers. For instance, Brian Storey wants to be able to "read the screen" from SuperBasic. Individual pixels can be read with PIXEL% (Volume D, *QL World* June 1989) but until now there was no way to read character codes from the display once they have been printed.

T. Ashcroft suggested a SCREEN$ function similar to the Spectrum's but added, "I realise that the different type sizes and windows might make this difficult." SCREEN$ is relatively easy on a Spectrum, even though it's bugged in all Spectrum ROMs and goes wrong if used part-way through an expression . . .

The Spectrum has just one character size, attribute graphics and convenient byte-oriented characters. Even so SCREEN$ will only read character-codes 32 to 127, and gets confused by underlining. The QL must do better.

Swedish Sinclair fan Orjan Larsson wrote to describe a game he has written, saying, "it would have been far easier if I could check which character is at a location. I can write machine code but must admit there are great problems doing it, compared with an IBM display that stores character codes, not pixels."

Orjan put his finger on the main problem: the QL PRINT routine converts character patterns from the fount into widespread pixel changes on the screen. A single character may end up spread through 60 bytes of video memory, with bits of other characters intermingled.

If that's not complicated enough, remember that windows may be anywhere on screen, with eight standard CSIZEs, two or three MODEs, stippled colours, FLASH, underlining, and re-definable founts. Text might be positioned at character, pixel or graphics co-ordinates using AT, or CURSOR with up to five parameters.

These variations make QL character recognition difficult, but they're trivial compared with real-world character recognition problems. At least we're not trying to read handwriting! To be really useful, CLIP functions must take *all* the QL options in their stride. Subset implementations are deservedly unpopular, because they make it hard to be sure whether or not a trick will work.

After a little thought I decided that standard CSIZEs were not enough. My CLIP functions should work with extra sizes, like the 7-pixel wide characters Psion use in their 64 column displays and the 9-pixel high text in Devpac's command window. CLIP should even work with *Minerva's* second screen, in case someone wants to copy characters between the two screens.

Several readers have suggested applications that need CLIP to work. Hugh Miall recently asked, "Would it be possible to develop a utility with which one could highlight or box any piece of text that is currently being displayed from any application, and then cut and paste it into another document or database in any other application? The Mac lets you do this cleanly by cutting and pasting to and from the clipboard."

Howard Clase made a similar suggestion last year, after using a BBC Micro. "SuperBasic scores over BeebBasic in most ways," he writes, "but the real challenge would be to mimic the BBC's COPY key, so that characters from the screen could be re-entered into the keyboard buffer by manipulating a cursor."

This suggestion was echoed by Brian McNulty, who wants a 'last line recall' routine. It is infuriating to have to retype a long line when you can still see the almost-right version on screen. *Toolkit 2* and *QPAC 2* include ALT ENTER, which recalls recent key-strokes, including arrows and single key entries, which may mess things up. A clipboard is a better answer, as long as it's easy to use. It can pick up the displayed line, or any part, without the irrelevant editing keystrokes.

A clipboard makes it possible to avoid copying and re-typing long file-names from directory displays; you should be able to 'scan' them into any keyboard queue. This saves time and reduces errors, and that's just the sort of routine I like.

If IBM's, Spectrums and Beebs can read characters from the screen, the QL should be able to do the same. Despite problems caused by the flexibility of the QL display, character scanning is too good a trick for a multi-tasking machine to be without.

Other tries

In some ways it is surprising that CLIP functions have not already been written. Quanta's library comes close with a function C_PIXEL, written by Ron Dwight in 1985.

C_PIXEL can discern characters anywhere on a screen, but it only recognises text from a preset 101-character set, in small MODE 4 letters on the standard screen, and may spend a minute or more scanning, if you start in the wrong place!

C_PIXEL is cleverly written but shows its age; it is not romable and won't compile with Turbo or Supercharge. The code is four times the size of my CLIP functions, and subject to a licence restricting Quanta members to 'non-commercial' use.

At first this sounded too big a job for DIY Toolkit, but after much experiment I developed a SuperBasic prototype that proved that characters could be read from the QL screen, regardless of fount, size, MODE or position. The prototype was rather slow, but it's easy to speed things up if you can afford the time to do it!

The main breakthrough was choosing the right intermediate data-structure. It's a big step from the QL display format, with 32768 interleaved colour bytes, to Ascii codes. I decided to use the QL fount data-structure as a staging-post.

Listing one is my prototype. The IMPLICIT% directives at the start identify integer values to Turbo. PATTERN$ holds the scanned fount image, and characters are read from a SCR window overlaid on the whole screen.

PICK_WINDOW uses DIY Toolkit CHAN functions from May 1988 to extract details from the QDOS Channel Definition. The magic numbers are address offsets, documented in Volume C. Nine calls yield the character-size, position and extent of the window. These values are used later by BUILD_CHAR_GRID and LOOKUP_CHAR.

BUILD_CHAR_GRID assembles a nine byte string that corresponds to the QDOS fount entry fro the character being scanned. QL characters are at least six pixels wide and ten high. The first row of

Listing one

```
100 REMark Prototype CLIP$ function for QDOS
110 REMark Copyright Simon N Goodwin 26/6/89
120 :
130 IMPLICIT% base_x,base_y,edge_x,edge_y,ch,x,y
140 IMPLICIT% cstep_x,cstep_y,pstep_x,pstep_y
150 IMPLICIT% row,col,background,slice,ch
160 :
170 DIM pattern$(9)
180 OPEN #3,scr_512x256a0x0
190 OVER #3,-1
200 :
210 PICK_WINDOW 2
220 REPeat loop
230   BUILD_CHAR_GRID
240   LOOKUP_CHAR
250   x=x+cstep_x
260 END REPeat loop
270 :
280 DEFine PROCedure PICK_WINDOW(ch)
290 LOCal attr%
300   base_x=CHAN_W%(#ch,24)
310   base_y=CHAN_W%(#ch,26)
320   cstep_x=CHAN_W%(#ch,38)
330   cstep_y=CHAN_W%(#ch,40)
340   edge_x=CHAN_W%(#ch,28)+base_x
350   edge_y=CHAN_W%(#ch,30)+base_y
360   fount1=CHAN_L(#ch,42)
370   fount2=CHAN_L(#ch,46)
380   x=base_x : y=base_y
390   attr%=CHAN_B%(#ch,66)
400   REMark Reject unless OVER 0, UNDER 0, FLASH 0
410   IF attr% && 143 : PRINT #0,"Bad PRINT type." : STOP
420   IF attr% && 64 : pstep_x=2 : ELSE pstep_x=1
430   IF attr% && 16 : pstep_y=2 : ELSE pstep_y=1
440 END DEFine PICK_WINDOW
450 :
460 DEFine PROCedure BUILD_CHAR_GRID
470 LOCal row,col,background,slice
480 BLOCK #3,pstep_x*6,pstep_y*9,x,y+pstep_y,7
490 background=PIXEL%(#3,x,y)
500 pattern$=""
510 FOR row=y+pstep_y TO y+9*pstep_y STEP pstep_y
520   slice=0
530   FOR col=x TO x+5*pstep_x STEP pstep_x
540     BLOCK #3,pstep_x,pstep_y,col,row,7
550     IF PIXEL%(#3,col,row)<>background
560       slice=slice !! 2
570     END IF
580     slice=slice+slice
590   END FOR col
600   pattern$=pattern$ & CHR$(slice)
610 END FOR row
620 END DEFine BUILD_CHAR_GRID
630 :
640 DEFine PROCedure LOOKUP_CHAR
650 LOCal ch
660 LET c=-1 : SEARCH_FOUNT fount1
670 IF ch<0  : SEARCH_FOUNT fount2
680 IF ch>=0 : PRINT #0,CHR$(ch);
690 END DEFine LOOKUP_CHAR
700 :
710 DEFine PROCedure SEARCH_FOUNT(base)
720 LOCal index
730 index=SEARCH_MEMORY(base+2,PEEK(base+1)*9+9,pattern$)
740 IF index : ch=PEEK(base)+(index-base+2) DIV 9
750 END DEFine SEARCH_FOUNT
```

Listing two

```
100 REMark Sinclair QL World HEX LOADER v 3
110 REMark by Marcus Jeffery & Simon N Goodwin
120 :
150 CLS: RESTORE : READ space: start=ALCHP(space)
160 PRINT "Loading Hex..." : HEX_LOAD start
170 INPUT "Save to file...";f$
180 SBYTES f$,start,byte : STOP
190 :
200 DEFine FuNction DECIMAL(x)
210 RETurn CODE(h$(x))-48-7*(h$(x)>"9")
220 END DEFine DECIMAL
230 :
240 DEFine PROCedure HEX_LOAD(start)
290 byte = 0 : checksum = 0
300 REPeat load_hex_digits
310     READ h$
320     IF h$="*" : EXIT load_hex_digits
330     IF LEN(h$) MOD 2
340       PRINT"Odd number of hex digits in: ";h$
350       STOP
360     END IF
370     FOR b = 1 TO LEN(h$) STEP 2
380       hb = DECIMAL(b) : lb = DECIMAL(b+1)
390       IF hb<0 OR hb>15 OR lb<0 OR lb>15
400         PRINT"Illegal hex digit in: ";h$ : STOP
420       END IF
430       POKE start+byte,16*hb+lb
440       checksum = checksum + 16*hb + lb
450       byte = byte + 1
460     END FOR b
470 END REPeat load_hex_digits
480 READ check
490 IF check <> checksum
500     PRINT"Checksum incorrect. Recheck data.":STOP
520 END IF
530 PRINT"Checksum correct, data entered at: ";start
560 END DEFine HEX_LOAD
570 :
580 REMark Space requirements for the machine code
590 DATA 450
600 :
610 REMark Machine code data
620 DATA "43FA01A834790000","01104ED27EFF6002"
630 DATA "7E00347900000118","4E92667053436666E"
640 DATA "2031E80054892D49","0058C0FC0028D0AE"
650 DATA "0030B0AE00346C52","203608006B4C2040"
660 DATA "45FA005476004FEF","FFF6224F70094E43"
670 DATA "4FEF000A226E0058","4A806A0422007000"
680 DATA "4A876B1E4A416B12","1381E80055892D49"
690 DATA "005833BC0001E800","60044271E8007801"
700 DATA "4E753381E8007803","4E7570FA4E7570F1"
710 DATA "4E7570FC4E757028","B09067F220280022"
720 DATA "6BF04A406BECD0A8","0026B068001E62E2"
730 DATA "4840B068001C62DA","32280022D2680018"
740 DATA "34280024D468001A","24680032EF4AD4C2"
750 DATA "3E3C8080082E0003","0034671408810000"
760 DATA "082E000200346704","3E3CC0C01E3C00C0"
770 DATA "3A7C008008280004","00426702DBCD7A01"
780 DATA "0828000600426702","7A0226417407C441"
790 DATA "E649D24138321000","E56CC84728447408"
800 DATA "D5CD320B7C007605","7807C8413001E648"
810 DATA "D04030320000E968","C047B04C67025446"
820 DATA "DC46D24551CBFFE2","12C651CAFFD443E9"
830 DATA "FFF7720018192468","002A6110670A2468"
840 DATA "002E6108670272F9","70004E75121A7000"
850 DATA "101AB81A660E7405","2649284AB70C56CA"
860 DATA "FFFC670C508A5201","51C8FFE870F94E75"
870 DATA "0828000000426606","1414B41366E6142C"
880 DATA "0001B42B000166DC","4E75000000000002"
890 DATA "FE5C05434C495025","FE5805434C495024"
900 DATA "0000",'*',37893
```

Listing three

```
100 REMark Program to clip text from a Quill 2.3 window,
110 REMark in 80 64 or 40 column mode; set COLS to suit.
120 REMark Copyright 1990 Simon N Goodwin, version 0.09.
130 :
140 LET cols=80 : REMark Try 80, 64 or 40
150 IF cols=64
160   OPEN #3,scr_448x160a34x56 : _XSTEP #3,7
170 ELSE
180   OPEN #3,scr_480x170a16x56
190   IF cols=40 : CSIZE #3,2,0
200 END IF
210 UNDER #3,1 : REMark Accept underlined text
220 :
230 OPEN #4,scr_480x20a32x236 : REMark Echo window
240 PAPER #4,2,0 : INK #4,7 : PRINT #4;"CLIPBOARD"
250 text$="" : REMark Clears any previous stored text
260 :
270 FOR y=(cols=40) TO 15+(cols=80)
280   LineLen=0 : Line$="" : c$=" "
290   FOR x=0 TO cols-1
300     AT #3,y,x : PAN #3,0,115
310     old$=c$ : c$=CLIP$(#3)
320     IF c$<>old$ OR c$<>" "
330       Line$=Line$ & c$ : LineLen=LineLen+(c$<>"")
340     END IF
350   END FOR x
360   IF LineLen=0
370     text$=text$ & CHR$(10) : REMark Assume ENTER
380   ELSE
390     text$=text$ & Line$ : REMark Continuous
400     IF Line$(LineLen)<>" " : text$=text$ & " "
410   END IF
420   PRINT #4,LineLen!Line$
430 END FOR y
440 PAN #3,0,116 : REMark Cursor off
450 PRINT #4;text$\"Scanned text is in TEXT$."
460 CLOSE #4 : CLOSE #3
470 STOP
```

pixels is always in the STRIP colour, separating one line from the one above. The next nine rows depend on the character pattern in the fount. Each row is controlled by a byte in the fount. Set bits give pixels in the INK colour, and reset bits correspond to the background STRIP.

Individual points are read with the PIXEL% function, from DIY Toolkit's Graphics Volume G. The BLOCK statements are not strictly needed, but illustrate how the prototype scans a character space. Much time is spent in system calls to address each pixel. The machine-code version is faster because it scans an entire character with just one system-call.

QDOS makes large characters by doubling up pixels horizontally or vertically, and adding extra columns of STRIP to the right side of the character. BUILD_CHAR_GRID uses different steps for larger CSIZEs, to ensure that it tests each distinct pixel from the pattern.

The scanned area is six pixels wide; this suits Sinclair founts and others intended for use in CSIZE 0,0 or 2,0 – whatever size they are eventually printed. Late roms can write eight pixel 'fat founts'. Small changes make BUILD_CHAR_GRID suit these; replace 6 with 8 in line 480, and 5 with 7 at 530, then make sure that SLICE still fits in a byte.

LOOKUP_CHAR uses the Turbo Toolkit function SEARCH_MEMORY to scan both founts for the pattern from the screen. SEARCH_FOUNT checks all eight bits, which is why SLICE is biased away from the two least significant bits, which are blank in Sinclair and Thor founts.

SEARCH_MEMORY is quite fast, but not ideal for this purpose; it tries every possibility, and may 'recognise' in-between patterns like the bottom of a 'g' and the top of an 'h'. It can't match underlined text, which requires that the eighth row of the character should be treated specially.

Faster

The new resident functions CLIP% and CLIP$ do not have these problems, and they're even faster; on my 640K QL the custom code scans rom founts at 710K a second. The prototype takes about a second to read a single character, even if the illustrative BLOCKs are removed. Contrast that with the machine code in **Listing two,** which can pick up between two and four 80-column lines each second, on a QL, and around seven lines per second on my Thor XVI.

The Basic prototype does a lot of unnecessary work. For instance the scanning loop from line 500 to 600 is emulated by just 20 machine code instructions, added to the display device driver. The design of the routines is almost identical, but optimisations and improvements in the machine code version make it over 200 times faster.

Listing two lets you enter the machine code without using an assembler. The full source of the CLIP functions will appear next month, with a commentary on the code and data-structures used. Listing two loads code from DATA statements, and saves it in a binary file. Once you've loaded that file, as follows, you can use CLIP% and CLIP$ in your own programs.

```
base=RESPR(450)
LBYTES "filename",base
CALL base : NEW
```

Listing two starts with the standard loader used in every month's DIY Toolkit project. Only the DATA, from line 590 onwards, changes from month to month. Line 150 uses *Toolkit 2's* ALCHP function to reserve memory for the data; the memory is recovered when you load another program. You must use RESPR instead of ALCHP on a machine without toolkits; alas, RESPR space is permanently allocated.

The functions have been tested using interpreted and compiled SuperBasic on QLs and Thor XVI. They run in ram or rom, and may be shared between any number of tasks. QPAC and swapper users may need to 'unlock' windows (POKE SYS_VARS+133,1 on a Thor); there's no problem with tasks controlled by DIY Toolkit utilities like *Taskforce* and *MultiBasic.*

The functions expect a single parameter – a SuperBasic channel number – rather than an optional plethora of pixel or graphic co-ordinates, fount addresses, CSIZEs, and suchlike. If you want these controls, you can use normal SuperBasic commands to set up the window concerned, or another that matches it.

Use AT, CURSOR or over-printing to step from one character to the next, and CHAR_USE, SET_FONT or _FOUNT to change the character_set. The DIY CHAN and CHBASE functions come in handy for finding the details of windows owned by other tasks.

The harsh # before the channel number is optional, but a pair of brackets and an integer value are compulsory. CLIP functions give the usual 'Bad parameter' and 'Channel not open' reports if supplied with faulty parameters, like unused, file or serial channel numbers.

There are two versions of CLIP, because some programs expect a numeric Ascii code and others prefer a character. CLIP% returns an Ascii code form 0 to 255, although CHR$(10) is never matched as ENTER does not make a pattern on the screen.

If anything goes wrong CLIP% returns a QDOS error code to indicate the problem. -1 means 'not complete' and indicates that the channel is already busy, usually waiting for input.

CLIP will not scan outside the chosen windows; CLIP% returns -4, 'out of range', if there is not room for a character at the current cursor size and position. -7 means 'not found': the pattern does not match any character in the founts. CLIP$ just returns a character, or nothing, which is often eloquent enough.

The following line shows CLIP$ in action. It clears the screen and lists itself at the top of the screen: then it moves a cursor over the listing, reading each character with CLIP$ and echoing it in window #0:

```
10 MODE 4: CLS: LIST #1: FOR x=0 TO 73: AT 0,x: PAN 0,115: PRINT #0, CLIP$(#1);
```

I got that program line into *Quill* by changing the PRINT #0at the end into a TYPE_IN, and running the result; PAUSE 150:RUN gave enough time for me to Control C to *Quill* from Basic. The program line literally **typed itself** into my document!

PAN 0,115 turns on the cursor in the default window, so you can see where the scan has reached; use y=10_TRAP(14) instead, on current Thors. Unlucky people with Minerva 1.62 or 1.63 might upgrade to 1.64 (my current choice), or use a Toolkit command like CURSOR_ON, CURSEN or whatever. PAN 0,116 turns the cursor off, on most machines.

This nifty example begs a few questions. It's easy to find characters in the Basic windows, because you have their

Incoming Mail

I'm excited about the prospects for DIY toolkit; I have some excellent ideas up my sleeves for future months, largely thanks to letters and suggestions from readers and friends. Further ideas are welcome, care of QL World. Next month I shall explain how to CLIP characters from any QDOS display, explaining the code and further applications.

DIY Toolkit programs are available in machine-readable form, for people who want further examples or wish to avoid the hassle of re-typing and checking the magazine listings. The new Volume S (for SCREENs) includes assembler source, Quill text, and examples too long to list here, including a multi-tasking clipboard that can read characters printed in any QDOS window and represent them to other tasks. Volume S costs £7 on disk or microdrive cartridge, with one free module of your choice for those who order before the end of January. The price includes disks and delivery, but microdrive cartridge orders must be accompanied by one formatted cartridge for each volume required.

14 Volumes of DIY Toolkit are now available, at £3 each, plus £4 per order for processing. You will find lots of new material on the disks, even if you have keenly read every column for the last three years. I have made many improvements, answered questions from those who used the original versions, and added examples that show the synergy between old and new DIY modules.

To obtain DIY disks, or a full list of volumes, ring Richard Alexander on (0559) 384574, or write to:

DIY Toolkit, Cwm Gwen Hall, Pencader, Dyfed, Cymru SA39 9HA.

channel numbers and can control them with AT, WINDOW, CSIZE, CURSOR and so on. It's more difficult to scan text from another task. You must get the cursor in just the right place, or a minus might be mis-read as an underscore, or – more likely – the pattern will not match the founts at all.

It could be difficult to line up the cursor above characters from other tasks; you need pixel precision to read characters correctly. An ideal 'clipboard' task should know the size and position of the window being scanned. It also needs the CSIZE or pixel spacing of characters, so it can step between them easily.

Next issue I shall present and explain routines to select and scan up to 32K at a time from any window, and transfer the results to any keyboard queue with QUEUE%, from Volume Q (of course). My latest DIY Toolkit opus includes a self-configuring Clipboard task, plus programs and text to accompany the CLIP code.

Listing three is a more substantial example; a generalised text clipper for Psion programs. It reads up to 32K into a string, TEXT$, from Psion's main data window, used for Quill text, *Abacus* and *Archive* data, in either 40, 64 or 80 column mode.

The program assumes the window locations used by Quill version 2.3, but should suit other variants. The 64 column setup uses the *Speedscreen* command _XSTEP to select characters seven pixels wide. DIY alternatives include POKE_W CHBASE (#3)+38,7 or CHAR_INC 7,10 for Toolkit 2 users. If confused, stick with the other two modes.

Underscores and underlined text might be a problem, as underlining changes the character pattern in screen memory. CLIP% and CLIP$ check the setting of UNDER, and ignore the eighth row of each character if UNDER 1 is set.

This means that underlined characters can be read and recognised as easily as Bold and Normal text. However underlining makes underscores and spaces indistinguishable, unless you re-define CHR$(95) of Sinclair's standard fount.

There is absolutely no difference between an underlined space and an underscore, as far as the screen is concerned, so CLIP functions cannot tell the difference unless you tell them the context. Set UNDER 0 if you want to be able to read underscores, and UNDER 1 to decipher underlined text, in which case underscores will be scanned as spaces.

Quill adds spaces to give neat margins, but there's no point scanning those as they were not part of the original text, and will be added (if necessary) when the scanned text is re-formatted. Listing three condenses clumps of spaces into a single space, but doubled spaces may appear if underlined words were missed because of UNDER 0.

PROCEDURE WHY

I got my toolkits in a twist in the October 1990 issue, substituting BASIC_L for BPEEK_L in PROC WHY. It works for me , and others with both Toolkits in their QLs, but I meant to use the DIY Toolkit function BPEEK_L. This loads with the other BPEEKs, or the BPOKEs in DIY Toolkit Volume B, currently top of the pops among DIY Volumes. DIY disk copies were fixed at the start of November, but earlier copies need:

```
200 rtstart=BPEEK_L (56)
210 rtend=BPEEK_L (60)
```

Sorry!

Accents

Accents, superscripts, subscripts and other fount variations are just as easy to read as long as you set the appropriate founts, or copy their addresses from Psion's window. Similar code can read characters from any window, but you may need to add lines to set new window positions and cursor sizes.

If you have the channel ID you can use CLIP to read characters directly from the task's window, avoiding problems with founts and alignment, but this can be restrictive, as QDOS does not allow simultaneous operations on one channel; you may be locked out if the window you want to read is waiting for input.

As you use CLIP% and CLIP$ you will probably find more and more applications for them. At last your computer can read the displays that you can – and accurately, too!

The Advanced Wordprocessor for the Sinclair QL

text87

text87 version 3.00 offers today's state-of-the-art user-friendly environment for document production. Extremely fast in operation with documents of any size and capable of mixing and displaying different type styles and sizes. With integrated spelling-checker in three languages and extensive support for highest quality printing possible on daisywheel and 9-pin printers.

text87 *budget*, has all the features of text87 version 3.00, the only limitations is that it cannot be used with 24-pin, laser and deskjet printer drivers. Upgrade to the full version will be availabe at the price difference.

text87: £60 ● text87budget: £45 ● fountext88 + founted89: £40
2488: £15 ● typeset90-deskjet: £20 ●
typeset90-laserjet: £40 ● typeset90-GQ5000: £40

Send for our new comprehensive leaflet

Software available in English, French, German and Italian. Prices are inclusive of airmail. Payable by cheque, Postal Order or Eurocheque. Please specify language and disk system (all 31/2" and 51/4" formats can be supplied). text87 requires at least 256K memory expansion.

Printer Drivers

fountext88 and founted89, the graphic printer-driver and editor for text87 provide graphic founts without the limitations in text editing and document size imposed by QL desktop-publishing programs. More than 35 quality founts of up to 72 pixels in height are supplied and new founts can be designed or captured from saved screen images. For Epson and compatible 9-pin and 24-pin printers.

2488, the state-of-the-art text-mode printer drivers for selected Epson LQ, NEC P, Star LC and XB and Panasonic KXP 24-pin printers offer advanced features such as multiple typefaces, proportional spacing, micro-justification, double height, shadow and outline modes (depending on the printer model).

typeset90-deskjet text-mode printer drivers for Hewlett-Packard Deskjet, Deskjet Plus and Deskjet 500 support printer's internal founts plus the full range of Roman and Helvetica plug-in cartridges.

typeset90-laserjet text-mode printer driver for Hewlett-Packard Laserjet series II and III printers supports printer's internal founts plus a wide range of fount cartridges (incl. Roman, Helvetica, Universe) in sizes between 6 points and 72 points.

typeset90-GQ5000 text-mode printer drivers for the Epson GQ-5000 supports printer's internal scalable and bit-mapped founts in sizes between 6 points and 72 points.

Software87, 33 Savernake Road, London NW3 2JU

SOFTWARE FILE

HOME BUDGET

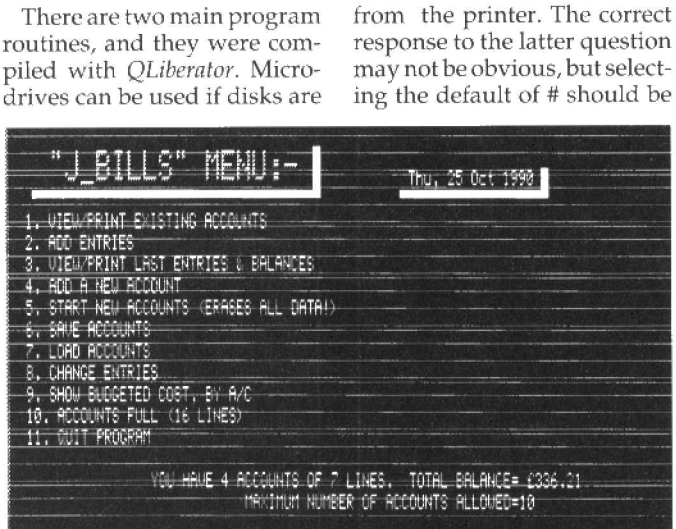
Home Budget has transferred to Dilwyn Jones Computing in a new, updated version. Bryan Davies tests his tax returns on it.

When *Home Budget* program was first reviewed (see *QL World* May 1990) it was in its initial version, which had several weaknesses. Unknown to us, an improved version already existed at the time the review was done (some months prior to publication) and it was only because the suppliers (PDQL at that time) did not send us an updated copy that readers were not given a view on the version they might actually buy. PDQL was by then already in a state where orders placed were not being reliably filled anyway. The author of the program finally got fed up with not being paid for his work and transferred the program sale rights to DJC, the present suppliers.

Revised

The program copy and instructions supplied for review initially were marked 'preliminary', but revised versions were supplied during the review period and only the instructions were then marked 'provisional'; there was a small updates file on the disk. One can assume they are representative of what buyers will now receive, since development of the program has been going on for a year or more. It is available on 3½in or 5¼in disk, and will run on a basic (unexpanded) QL. The author states that it is compatible with the two-screen mode of the Minerva V1.82 rom. The program is started from the Boot routine by EXEC_W, not EXEC, so cannot be multi-tasked with others. The instructions are on paper, rather than the usually-inconvenient disk files which are typical of programs in this price range. As the data tables used by the program can change from year to year, it is DJC's hope that yearly upgrades can be provided to registered users, at a small cost.

There are two main program routines, and they were compiled with *QLiberator*. Microdrives can be used if disks are not available. One feature retained from the original is the advice to set printer margins, or tell the printer how to produce a £ sign, through a few lines of SuperBasic before running Home Budget. This can be done quite simply in the Boot routine, but no doubt many users will not feel up to writing the necessary lines of SuperBasic and will be irritated by this feature. The initial menu does allow you to set the printer port and Baud value, and nominate a keyboard character which will produce the £ from the printer. The correct response to the latter question may not be obvious, but selecting the default of # should be successful with many printers (not with either of mine!); if you use an ESC-code sequence to produce the £, you can't enter this via the program menu. A working copy of program files can be made with either the supplied Clone routine or the COPY or WCOPY command.



The options screen on Home Budget.

The clone routine doesn't insist on formatting the working copy medium. It didn't work on my system the first time, stopping with a message apparently indicating that the master disk was write-protected. My first action on receiving program disks is to set the tab to protect, to avoid the risk of the master files being lost through some unanticipated error, and I don't like routines which write to the master disk. On re-trying the routine, and entering the device names with an underscore after them, it worked fine. Don't believe error messages implicitly! The instructions didn't point out the need for the underscore. Although the clone process worked when the target cartridge was formatted by the routine, it stopped when asked to make the copies without formatting the cartridge first. This turned out to have something to do with the fact that other programs were loaded in the review system, as the problem didn't occur when the clone routine was run on its own. The simplest approach is to use the WCOPY command (if you have it available) instead of the clone routine.

INFORMATION
Program: *Home Budget* V2.11
Price: £20 incl. EC postage (extra outside EC)
Supplier: Dilwyn Jones Computing, 41 Bro Emrys, Tal-y-Bont, Bangor, Gwynedd LL57 3YT. Tel. (0248) - 354023

It may seem picky to comment on apparent weaknesses when they are not in areas vital to the functioning of the program, but one's feelings about any program are influenced by such things, trivial though they may seem. On starting the program proper from mdv1, it expected data files to be on flp2_, even though the program default had been set to mdv1_; the Updates file indicated that this fault had been corrected, but that wasn't the case with the review copies. Over-typing flp with mdv and leaving the displayed drive number and underscore resulted in the medium not being found, but it was found when the number and underscore were retyped after the mdv. To be fair to the author, it is stated on the first page of the instructions that the whole alternative input needs to be typed in, but this comment may not register with everybody at first, and it would not have been a major job to remove the existing default device details from the display when the user begins to type a new default in.

Familiar

Users of the original version will find the latest one almost completely familiar. The menus, and the required responses, are essentially the same. Rather than repeat all that was said in the initial review (May 1990 issue of QL World), I will comment mainly on the changes that have been made.

In operation, the program uses about 60-70 KB of memory space, with the demonstration files loaded. The 'on-screen' free memory read-out lets you know how much further there is to go, as you add data. The demonstration budget file is set up to allow 10 accounts (expenditure categories), which is what a basic QL will allow. Roughly 3 KB is required per account. If you want to record more accounts, you need to rename your accounts files; for example, the demonstration file is named BILLS10_MAY and allows up to 10 accounts, with the limit being raised to 15 if you rename the file BILLS15_MAY. The main menu for the Bills section of the program is illustrated, as it is displayed when the demonstration file has been loaded.

Dates

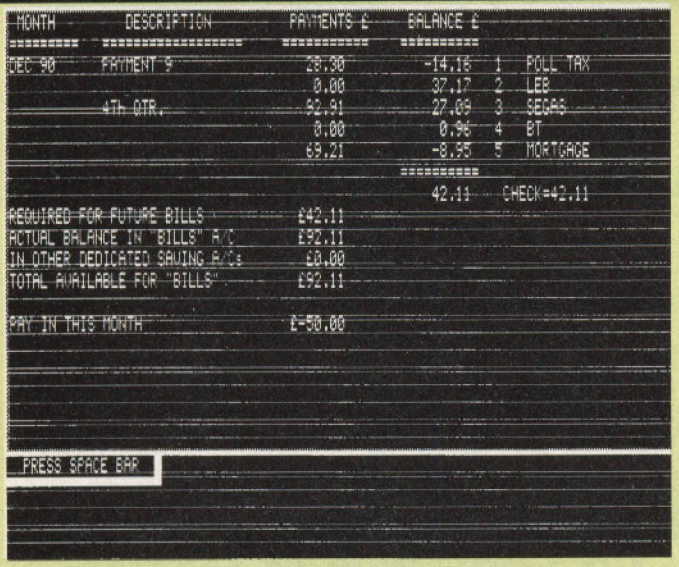
You select the Start New Accounts option to begin entering your own data. The next step is to enter account names, descriptions and amounts, month-by-month. Beware of making this selection immediately after viewing the demonstration file, because the program apparently remembers the dates used in that file and the first month now offered is November 1988 and the last is February 1990. Even if you start straightaway by selecting the Start New Accounts option immediately after booting-up, the dates displayed are November '89 to February '91, suggesting the program hasn't been updated in this area recently. It doesn't appear to be intended that you enter historical data.

As in the earlier version, the name of the account and the calculated monthly payments overwrite the identifying Line number, but this creates no problem. The first entry is displayed on Line 1, against November '89, but the second entry skips to October '90, the current month, on Line 12. You are then offered the option to make further entries, which you would not normally want to do at this point, assuming you are starting anew at the current month.

Summaries

Once you have finished entering account details and ask for the summaries to be displayed (the View/Print Existing Accounts option), the months November '89 to September '90 are displayed with zero entries, but the current month (October '90) contains the data you entered initially. Incidentally, there seems no point in numbering the main menu options, since you cannot select them by pressing the number keys, the up/down cursor keys plus ENTER being needed for that. After the first

```
TAX OF:- U. N. Representative
25 Oct 1990
INCOME TAX 1989/1990

 SLICE  RATE %   TAX   CUM.TAX  INCOME
 =====  ======  =====  =======  ======
 20700    25    5175    5175    20700
ABOVE #20700 TAX RATE IS 40%

ALLOWANCES:-

SINGLE PERSON'S ALLOWANCE   = #2785
MARRIED COUPLE'S ALLOWANCE  = #4375
WIFE'S EARNED INCOME ALLOW. = #2785
OR OR WIFE'S EARNED INCOME, IF LESS.
AGE ALLOWANCES - CONSULT TAX OFFICE

INPUT SUMMARY.
==============
EARNED INCOME              #21200.00
TAXABLE BENEFITS            #1200.00
UNTAXED INTEREST&DIVs        #200.00
BANKS&B.SOCs pre 6.4.91      #886.00
DIVs,INTEREST,ANNUITIES      #174.00
UK TAX CREDITS                #58.00
INCOME TAX ALLOWANCES       #5575.00
PAID TAX ON EARNINGS        #3906.00

REPORT
=======

EARNED INCOME              #21200
GROSSED INVEST. INCOME      #1613
TAXABLE BENEFITS            #1200
                           ==========
TOTAL GROSSED INCOME       #24013
less ALLOWANCES             #5575

TAXABLE INCOME             #18438
TOTAL TAX DUE               #4609
less:-
PAID TAX ON EARNINGS       #3906
DEDUCTED @ COMP.RATE        #295
UK TAX CREDITS               #58

ADDITIONAL TAX DUE           #349
NET INCOME AFTER TAX       #18205
MARGINAL RATE 25% (ON #18438)
AVERAGE RATE  25%
```



An account summary for the month of December

stage of opening accounts, you can then select the Add Entries option.

The View/Print Last Entries and Balances option lets you see whether or not the bank account balance you keep for paying the bills is positive or otherwise. A Pay In This Month line tells you how to much to put in to regain a zero balance, if you have 'gone over the top'. One point to note when making changes to estimated annual outgoings for any of the accounts set up is that any change made part-way through a year will affect the monthly savings figures only from the month of entry; that is, if your estimated gas bill for a year increases now from £500 to £700 pa, the monthly savings required will still be based on 500/12 for prior months, and on 700/12 from the current month on. There will be a shortfall (in this case) in the savings figure for the whole year, the size depending upon when, in relation to bill-payment dates, the annual figure is changed. You can make the amount up when you check the Last Entries and Balances. The calculation of the required figure is detailed in the second illustration.

Prompts

When you come to save a month's data, cautionary prompts should prevent you overwriting or deleting files which you still need. A nice touch is the option to save again, so that you can (conveniently) make backup copies. After entering 16 Lines of data, the program has to delete some of them to make room for new ones, and you are offered the option to save the current month's data under the existing file name with the suffix _FULL added to it; having offered this, the program refused to accept it for test file however, indicating that it was an unacceptable file name.

Tax idea

The tax-calculation part of the program is intended for individuals rather than businesses, although self-employed people having odd income from several sources (eg computer journalists) might be able to make use of it. It is pointed out in the instructions that the results may not agree exactly with those of the Tax Inspector. That is, use it to get an idea of where you stand, not as a basis for argument with the tax man. Subjectively, this part of the program works faster than the other. To the best of my recollection, this part also seems to have had more (obvious) attention paid to it since the earlier version was reviewed. While the entering of data is quite straightforward, deciding how certain income sources should be classified and treated is not something everybody is knowledgeable enough to do, and the instructions advise you to seek professional guidance on such income. On the other hand, general publications like "*Which*" magazine do provide enough information to enable the average earner to input the necessary income data correctly. The program instructions give a fair amount of explanation on how different types of income are handled. The third illustration is of the screen display after the demonstration tax file has been loaded and the See Report/Amend Tax Inputs option selected.

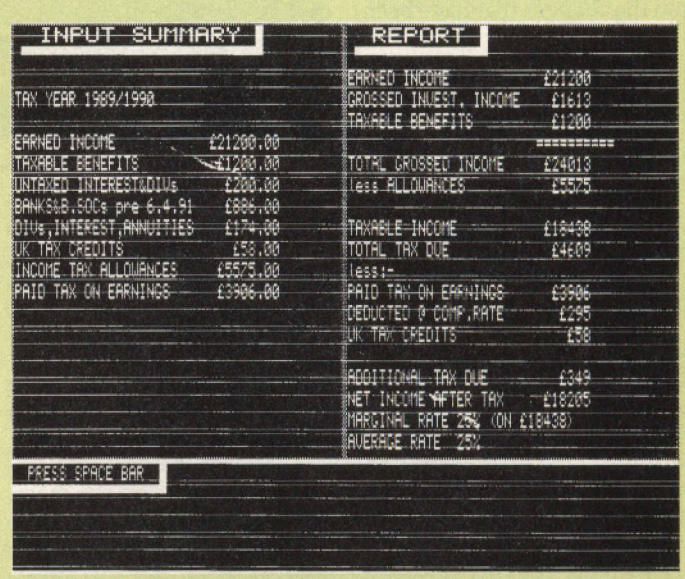
At all stages of the program, provision is made for hard copy to be made of the data. As mentioned earlier in this review, you are advised to send formatting information to the printer via Super Basic, before running the program, and that is by no means an unusual requirement with non-major programs, but it is an irritation. In this era of glossy presentation, it seems out-of-place to print documents with no left-hand margin, and in the least-appealing typestyle (non-proportional 10-pitch font), but that is what you get under the general heading of 'draft print'. The fourth illustration is of a tax statement, as printed out using the same menu option as for the third illustration (this was laser-printed, hence the lack of "dottiness"; note the # signs instead of £s).

A RPI (Retail Price Indicator) table is provided, and the user can update this information given in newspapers. There is a CGT (Capital Gains Tax) Indexation routine, which uses the RPI table; from this, you can find out the value of assets, indexed to reflect inflationary effects between the time of purches and that of disposal. Changes to the parameters and rules of Income Tax at Budget time can be entered, and saved, to keep the program operation up-to-date. DJC hope to provide users with updated files for the information which varies from year to year; the charge for this service is not quoted, but it should be small.

There have been some improvements to the program since the first review, and it will do a reasonable job of keeping you aware of the balance between regular expenses and your provisions for them.

Impending

It will also keep you aware of impending tax bills, if you are a person who has income not subject to PAYE deductions. There are still several areas in which the user interface could be more consistent, and data input categories described more clearly, but theses are not of fundamental importance to the 'works' of the program. The price is reasonable, and it is clear from the fairly regular mentions of home financial programs in letters, and in conversation, that there is a market for something like Home Budget; that is, for something simple and not cluttered with the type of features that are no use to the average user.



A tax summary for the year 1989/1990.

SUPER BASIC

In the final instalment of the arcade game program, Mike Lloyd brings action to the scenario produced by last month's routines.

Last month's *SuperBasic* article described the code required to set the scene for the Space Marauders arcade game, including its screen displays, text management, suitable noises and a score-keeping routine. This month's articles complete the program and describe the main game loop, the controls for the Space Marauders themselves, and code to manage the inevitable destruction of protagonists. Once these additional routines have been appended to last month's listings you have a fully-fledged Basic game which should reward the time spent typing it in with hours of enjoyment.

The heart of any arcade game is the 'game loop', a rapidly cycling length of code which manages all of the objects on the screen and responds to all keypresses. The design of this loop is a vital factor in the success of the program. The overriding consideration is almost always speed, even at the expense of program readability or good style. In the simplest of game loops, everything is checked, moved or calculated once in each cycle. Thus, the Space Marauder moves one jump, the Alpha Skimmer moves one space (if appropriate), collision detection is checked once, each key press is checked once, and so on. For more complex games some of these actions might be programmed to occur twice or more in each game loop, and other actions might only occur once every two or three cycles of the game loop. Basic interpreters rarely have the power to cope with such sophistications without unnecessarily slowing down the action, and so it is wise to stick to a straightforward game loop in which everything is checked and moved just once.

To plan a game loop you must list everything to be accomplished on each cycle and then decide on the optimum order for things to be managed. For Space Marauders, the list was:

```
IF right-keypress THEN
    adjust Gun location
IF left-keypress THEN
    adjust Gun location left
IF space-bar pressed THEN
    Fire routine
Draw Alpha Skimmer
Erase Alpha Skimmer
Calculate new Space Marauder position
Draw Space Marauder
Erase Space Marauder
IF Alpha Skimmer is hit THEN
    Hit routine
Update scores
IF no laser power THEN
    End game
IF no lives left THEN
    End game
```

Uppermost in a programmer's thoughts must be the problem of making objects appear to move smoothly when in fact they are being drawn in one location, overwritten, and drawn in another location in jumps. To fool the player into thinking that objects are moving smoothly, they must be allowed to remain at each location on the screen for as long as possible before quickly being erased and redrawn somewhere else.

Overdrawing

There are two ways of overdrawing objects on the screen: the area they occupy can be overprinted by a blank character or block of the same colour as the background, or else the object can be drawn again but using the OVER command with the -1 parameter. OVER -1 changes any ink colour to the logical opposite of the paper colour it is overprinting, so printing something for a second time reverses the initial printing without affecting the background it was printed on.

This is most easily explained by using black paper and white text as an example. When OVER -1 is selected and a black 'paper' pixel is changed to a white 'text' pixel the result is a white pixel, as might be expected. However, if 'white' text is placed on a background which is already white, then the two colours inter-react to produce a black pixel. Things can become slightly more difficult to follow when other colours are printed. If a red 'text' pixel is printed on a black background the result is red text, but on a white background the result is green text, green being the logical opposite of red.

The big advantage of using OVER -1 is that you can print and erase objects on backgrounds containing shapes of all colours without the object ever becoming invisible and without the background being steadily erased. As a bonus, printing with white 'ink' is slightly faster in OVER -1 mode.

There is an occasion when simple erasure of the object can be preferable to printing in OVER -1 mode. If the object is a character drawn on a regular background and it moves at most only one print position left or right with each game loop cycle then it can quickly be printed flanked by a space on either side. When printed, this string will always remove all evidence of its predecessor no matter which way the central character moves. Printing a string of three characters is almost always faster than printing a single character twice, except on the QL due to its strange print management algorithm. This is not the case when *Speedscreen* or *Lightning* is used, but nevertheless for the sake of users of standard QLs the string method has not been used in the Space Marauders game.

The rationalised list of occurrences for each cycle of the game loop can be read directly from **Listing seven,** but a precis is:

```
Erase Marauder
Calculate new location for Marauder
Draw Marauder at new location
Draw Alpha Skimmer Gun
Detect keypresses:
    IF space-bar THEN
        fire laser, etc
    IF left cursor THEN
        erase, move and redraw Gun
    IF right cursor THEN
        erase, move and redraw Gun
IF Marauder close to Gun THEN
    destroy Gun routine
    and exit game loop
IF no laser THEN
    Exit game
```

Unusually, the main game loop has a strictly controlled number of iterations and it is inside another loop. This is because the track of a Space Marauder across the screen takes a set number of game loops to complete. Most games last for an indefinite time until something occurs which forces a jump out of the game loop and so a REPeat loop structure is usually preferable to a FOR . . . NEXT loop.

Ellipse

The Space Marauder is an ellipse which follows a figure-of-eight orbit. The parameters for the orbits are contained in the DATA commands in **Listing eight** and are selected by choosing a RESTORE value at random. Each orbit begins somewhere just outside the screen picture towards the top of the screen. The Marauder will then curve down towards the bottom of the screen where it will momentarily disappear 'behind' the Alpha skimmer before its orbit takes it once more to the top of the screen. To enhance the three-dimensional feel of the game the size of the ellipse increases as it moves towards the bottom of the screen.

The algorithm for the orbits is based on those for Lissajous curves, introduced to *QL World* readers many months ago as a way of producing interesting screen patterns. Unlike most home computers, where rapid screen drawing outperforms mathematical ability, the QL is not very fast at updating screens but is much better at rapid mathematical calculation. It is therefore acceptable to put some quite complicated mathematics inside the game loop of a QL game.

The location of the Alpha Skimmer (or Gun) is determined by the variable XX which specifies its horizontal displacement. The CURSOR command is used in preference to the PRINT command so that the traversing speed of the Skimmer can be adjusted to an optimum value. To avoid having to define a new character to represent the laser gun, Ascii character 162 is employed as a graphic.

Keyboard activity is detected using the KEYROW function to minimise the time spent detecting and acting on a keypress and so that two simultaneous keypresses can be responded to in the same game loop.

Explanation

The IF statements which analyse keypresses need some explanation. The && operator ine of a set of binary operators which act on the bit patterns of any value. As you will know, the QL's internal model of the world is represented entirely in terms of the 1s and 0s of the binary counting system. Binary digits are handled in groups of eight, each known as a byte.

LISTING 7

```
700 DEFine PROCedure game_loop
702   REPeat game
704     RESTORE 800 + RND(1 TO 12)
706     READ y, z, start, finish
708     FOR n = start TO finish
710       CIRCLE xps, yps, (100-yps)/10, .5, 1.6
712       xps = 85 +65 *SIN(z+ 20*PI* (20+n)/300)
714       yps = 50 +60 *SIN(y* PI*(20+n)/300)
716       CIRCLE xps, yps, (100-yps)/10, .5, 1.6
718       CURSOR xx-3, 10, 0, 0
720       OVER 1: PRINT CHR$(162);: OVER -1
722       k = KEYROW(1)
724       IF k&&64
726         noise 2
728         FOR f = 1, 2: LINE xx, 10 TO xx,power
730         IF ABS(xx-xps) <5 AND power >=yps
732           hit: EXIT n
734           ELSE : power = power-5
736         END IF
738       END IF
740       IF k&&2
742         CURSOR xx-3, 10, 0, 0
744         OVER 0: PRINT " ";: OVER -1
746         xx = xx - 10*(xx>8)
748       END IF
750       IF k&&16
752         CURSOR xx-3, 10, 0, 0
754         OVER 0: PRINT " ";: OVER -1
756         xx = xx + 10*(xx<156)
758       END IF
760       IF yps>-2 AND yps<20 AND ABS(xps-xx)<20
762         die: EXIT n
764       END IF
766       IF power <0: EXIT game
768     END FOR n
770     IF lives <0: EXIT game
772     scorex
774   END REPeat game
776 END DEFine game_loop
```

LISTING 8

```
800 REMark >                              Saucer Control
801 DATA  9,  0,  4, 56
802 DATA  7,  9, 10, 80
803 DATA 16,  1, 30, 62
804 DATA  6,  2, 18, 94
805 DATA  8,  4,  8, 66
806 DATA  9,  8,  6, 56
807 DATA 14,  0, 36, 74
808 DATA 15, 17, 30, 66
809 DATA 15, 27, 36, 67
810 DATA 25, 25, 10, 33
811 DATA 20, 15, 20, 45
812 DATA 25,  9, 12, 32
```

LISTING 9

```
900 DEFine PROCedure hit
905   pts = pts +10: power = power +10
910   noise 6
915   FOR blast = 0 TO 16 STEP 2
920     INK blast
925     FOR n = 1, 2: CIRCLE xps, yps, blast
930   NEXT blast
935   INK 7: BEEP
940 END DEFine hit
```

The && operator looks at two values and returns a 'true' value if the '1' bits of the second value are also set to '1' in the first value. The binary value of 11 is '1011'. The expression 11 && 4 returns a 'false' value because in binary the value 4 is '0100' and the bit set to '1' here is set to '0' in the binary number for 11. The expression 11 && 8 returns a true value, though, because eight is '1000' in binary and both numbers have their first bit set to '1'.

Keyrow

The KEYROW function returns a value between 0 and 255 which can be decoded to find out which keys in a particular row of the KEYROW table have been pressed. Read your User Guide for more details of how KEYROW works. For now, it is important only to know the following line from the KEYROW table:

```
ENTER   =   1
LEFT    =   2
UP      =   4
ESC     =   8
RIGHT   =  16
\       =  32
SPACE   =  64
DOWN    = 128
```

If the Enter key is pressed, KEYROW(1) returns 1, if the right cursor key is pressed then KEYROW(1) returns 16, and so on. If both the ESC key and the down cursor key are pressed simultaneously, then KEYROW(1) returns the sum of the two values, in this example 136.

The && bitwise operator is used to compare the value returned by KEYROW(1) and a constant to discover if the key represented by that value has been pressed. The expression KEYROW(1) && 2 is true whenever the left cursor key is pressed, even if other keys are being pressed at the same time.

Laser fire

The actions taken on the detection of a valid keypress are much more straightforward. When the laser is fired, two lines are drawn vertically from the current position of the Alpha Skimmer. The height they reach depends upon the strength of the laser. Because the screen is in OVER -1 mode the second line overwrites the first, making the laser beam appear only momentarily. If the centre of the Space Marauder is within five graphics units of the laser beam it is deemed to be destroyed and its energy tops up the laser beam's power supply. The procedure which handles this sequence lies outside the game loop because speed is no longer important until the next part of the game begins. If the shot misses then the laser's strength is reduced by five to discourage rapid automatic fire.

If the left or right cursor keys are pressed, the Alpha Skimmer is erased at its present position, its next location is calculated and it is drawn again so quickly that it appears to have slid to the left or right. Checks are included in the algorithms to prevent the Skimmer moving too far to the left or right of the screen.

LISTING 10

```
1000 DEFine PROCedure die
1005   FOR x = 1 TO 7
1010     INK x: LINE xps, yps TO xx, 5
1015     CURSOR xx-3, 10, 0, 0: noise 6
1020     PRINT CHR$ (RND (180 TO 191));
1025   NEXT x
1030   lives = lives -1
1035   CURSOR xx-3, 10, 0, 0
1040   OVER 0: PRINT " ";: OVER -1
1045   xx = 86:  BEEP
1050 END DEFine die
```

LISTING 11

```
1100 DEFine PROCedure wow (word$)
1105   OVER#2, -1
1110   FOR x = 1 TO 20
1115     AT#2, 2, 22: PRINT#2, word$
1120   END FOR x
1125   OVER#2, 0
1130 END DEFine wow
```

Destruction

The locations of both Skimmer and Marauder are compared to see if the player has allowed them to come too close together. If this has happened, control leaps from the game loop to a procedure which handles the destruction sequence and the game loop is left.

If there is no laser power remaining, then there is no point in the game continuing, so towards the end of each game loop a check is made on the laser power value and the game loop is terminated if the gun can no longer be fired.

The game loop terminates only if a Skimmer has been destroyed or if its laser power is used up. On such occasions the number of lives is checked to see if the game is completely over. If the game is to continue, the Scorex routine is called to update the player's score. At the end of a whole game control returns to the main loop described and listed last month.

Listing eight contains 12 settings for Marauder orbits. The line numbering is critical because it relates directly to the RESTORE statement. Each setting comprises 3 values used directly in the algorithms for the orbits and a fourth value which defines the number of iterations of the game loop required to complete one orbit. You can experiment with changes to the values, but be warned that quite small changes to a parameter can have dramatic changes on the paths followed by the Marauders, many of which will not be suitable for the game.

The ninth listing manages the actions which accompany a hit on a Marauder from the laser. Both the laser power and the points scored values are increased and the Marauder is blown to smithereens with a suitable noise. The 'noise' routine was listed last month. The chosen noise will persist permanently unless another BEEP command is issued.

Extra life

The sequence required when an Alpha Skimmer is hit by a Marauder is detailed in Listing 10. A coloured line of plasma links the Marauder and the Alpha Skimmer, which then disintegrates by being overwritten by a random series of characters. Before starting the next game, the XX variable is set to place the next Skimmer roughly in the centre of the screen. Once again, a BEEP command is needed to silence an otherwise permanent noise.

The final routine is used when an extra life is won, as determined by the Scorex routine from last month. The text 'Extra Life!!' is flashed repeatedly on the screen. The listing screen, which occupies the same screen space as the main window, is used for printing this message so that the Skimmer's printing position is not disturbed.

The Space Marauders game has just about enough action, novelty and skill level to make an entertaining game. It can be compiled for additional speed, but the best results are obtained from running it in SuperBasic with the Lightning graphics routines loaded. Of course, a compiled version running with Lightning will be even faster.

Achivist Robin Stevenson opens Archive up for newcomers.

ARCHIVE: the Absolute Beginning

Maybe you are new to the QL and have not yet looked beyond *Quill*. Or perhaps you have been meaning to get round to looking at Psion *Archive* for the last few years, or you occasionally stumble across the Archive section of the manual by mistake. Well now is the time to locate your Archive microdrive and finally have a go.

The first requirement, as with all software, is to make sure you use a copy, rather than the original, in case it gets damaged or corrupted. Psion include a program to do this, which can be run from SuperBasic. With the original program in mdv2_ (the right hand one), and a blank cartridge in mdv1_, you type LRUN MDV2_CLONE<enter> and the program will be copied on to the new cartridge.

To run Archive, like all the Psion programs, you normally start by resetting the computer. Then with the copy of Archive in the left hand mdv1_ slot, and a formatted cartridge for the data in the mdv2_ slot, select F1 monitor, or F2 tv. Archive will run automatically and give a screen that looks very much like all the other Psion programs.

If you have disk drives you will need to alter the SuperBasic program called BOOT to read EXEC_WFLP1_ARCHIVE. You will also need to run the CONFIG_BAS program found on the *Abacus* cartridge. This configures the program, so that it uses the disk instead of microdrives for such things as data and help files. Humble microdrive users need not worry about any of this.

Edit

The major difference in use between Archive and the other Psion programs is that it is not menu driven. To use an option you have to type the command in full. Furthermore, the commands available can be added to by writing programs, in the Archive programming language. This is what we are going to do now. Type EDIT<enter> from the prompt. You are now in the Archive editor, and where it says 'proc', you must enter the name of the new procedure. Type in NEWINDEX;NAMES<enter>. This is the first line of **Listing One**. Carefully type the other lines, finishing each with an enter. You should not type in the last

```
Listing One

proc NEWINDEX;NAME$
  create NAME$, logical "INDEX"
    WORD$
    SUBSID$
    PAGE
    endcreate
  endproc
```

line, that says 'endproc'. Archive includes this (like the 'proc' at the start) automatically.

Don't worry about the mixture of capitals and lower case letters; they are all treated as identical by the Archive program. The *exception* to this is single and double quote marks. Archive will accept either, but both of a pair must match. While typing each line all the normal text editing keys can be used. Once all the lines are entered, press ESC to finish inserting. If you now notice some typing mistakes, you will need to go back and alter these lines. The information box, 'Using the Archive program editor' explains how to do this. It also explains other facilities available in the editor, which you could experiment with. When the procedure is complete and correct, press ESC again to return to the normal Archive screen.

The procedure tells Archive to create a data file, and specifies the structure of that data (of which more shortly). It could have been entered directly from the normal Archive prompt. Two advantages of writing it as a program are that you can edit it until it is exactly right; and once it is right you can use it many times.

The example program I have chosen is to create an index for a book. It stores all the words and page numbers as data, and then produces a properly ordered index, just the thing for your seminal work on Euclidean geometry, WI cook book, or whatever. The data is to be stored in files created by NEWINDEX, so called because it creates a new data file each time it is used. These are automatically stored on microdrive, using a file name which must be different for each index. Each time NEWINDEX is run, the name must be supplied as well, as a 'parameter'. This is typed after the procedure name, separated by a semi-colon.

Testing

To test the procedure, its name is typed at the prompt, along with the file name inside quote marks. Type NEWINDEX; "EUCLID"<enter>. You will hopefully hear a whirring of microdrives, and then the cursor will re-appear. If you get an error message, it means you made a typing mistake. You must now decide if the error was in the program, or just in your test line. If you can see a mistake in the test line, call it back for editing by pressing F5. Correct the mistake, and press enter, to try again. If the error is in the program, it will write out the line that confused it. You must now go back into the editor, by typing EDIT<enter>, and re-check the procedure before trying to run it again.

```
Listing Two

proc OPENINDEX
  local KEY$,FILE$
  cls : print "[O]pen an existing file, or [C]reate a new one ";
  let KEY$=upper(getkey())
  print KEY$
  input "Enter file name : ";FILE$
  if KEY$="C"
    NEWINDEX;FILE$
    else
    open FILE$ logical "INDEX"
    endif
  endproc
```

Once NEWINDEX has run without error, you will have created a data file. Prove this by typing DIR""<enter>, which will show a directory of your data drive, including the file "EUCLID_dbf". The _dbf extension is added automatically by Archive, and stands for 'DataBase File', showing that it holds data, rather than, say, program lines. When Archive creates a _dbf file, it also OPENs it. A data file can only be used within Archive while it is open. Show that EUCLID is open by typing DISPLAY<enter> at the prompt. This shows the current open file (if any), and shows its structure.

Archive cannot deal with long streams of text, or haphazard jumbles of data. The data it holds is organised into 'fields'. Each field is a separate item, and can be either a string of text characters, or a number. The field structure is fixed at the

time the file is created, and each record added must be fitted into the existing structure. Text fields must always end in a "$" sign, pronounced 'dollar' (WORD-dollar, SUBSID-dollar, etc.).

So far we have not added any records to the index file. It has a field structure, but no data. Despite that, we will CLOSE the file for the time being. To do this, just type CLOSE<enter> at the prompt. It is vital that you never remove a microdrive or disk that has open files on it, If you do, you risk corrupting the whole file, if not the whole disk. Always close all data files before changing the cartridge.

Whenever you leave Archive (by typing QUIT<enter>), you should not only ensure the data files are closed, but also that the program you are working on is 'saved'. Program lines are held entirely in the computer memory, so unless you transfer them to microdrive, they will be lost when you quit. Archive stores all the procedures together as a single program. We shall use the name 'indexer' (file names cannot be more than eight letters long), and can save it by typing SAVE "INDEXER"<enter>. The resulting file is called INDEXER_prg, where the _prg extension shows it is a program file. A previously saved program can be LOADed back into the computer's memory. If you have not got time to do this tutorial all in one go, you could close the file, save the program, and quit from Archive. When you return to Archive you recall the program by typing LOAD "INDEXER"<enter>, and carry on where you left off.

However, rather than quit at this juncture we will press on with another procedure. This one is to be called OPENINDEX, and is to handle the choice between opening an existing index file, and creating a new one. Three elements are needed: we need to ascertain which of the two options is required; what the name of the file is; and to take appropriate action depending on the option chosen. **Listing two** shows this procedure, and we shall look at how it meets these needs.

To show the choice available we use the PRINT command to write the available options on the screen: to press C for create, or O for open. This key-press is captured using a function called 'getkey()'. A function is like a command, except that it has an answer, and the answer can be stored in a 'variable' for later use. In our case the key pressed is stored in a variable we choose to call KEY$, by using the line 'LET KEY$=GETKEY()'. We actually combine another function, 'upper()' which coverts letters from lower to upper case. By putting getkey() inside upper(), the answer from getkey() gets passed to upper(), and the answer from upper() gets put into KEY$. This means we can be sure KEY$ is a capital letter, regardless of what is typed.

Having got the answer to the first question, (and printed it back, just for show), we need to get the file name. Instead of getkey() we use 'INPUT', which can read whole words or lines. Like LET it assigns the answer to a variable. It can also include instructions as text, just like the PRINT statement. The INPUT command prints the text, reads in the response, (which must be finished with an <enter>), and stores it in FILE$.

Finally we have to act on the choice. In Archive, as in most computer languages, this is done with an 'IF' statement. In effect, this tells the computer: if x is true, do a particular set of actions. You can also specify what is to be done if x isn't true, by adding an 'ELSE' clause. Finally 'ENDIF' marks the point at which both branches rejoin. The IF statement is vital to programming, because without it a program would always do exactly the same thing, without any variation.

In this procedure we use the value in KEY$ as the test. If it is a 'C', NEWINDEX is called, passing the value in FILE$ as the parameter. If it is not 'C', then the Archive 'OPEN' command is used to open an existing file. Notice how our first procedure is called as though it is part of the Archive language.

To enter the new procedure, you again need to enter the editor. This time however, you are not automatically prompted for a new procedure name because there is already one there. To create a new procedure you press F3 N, and enter the new name, 'OPENINDEX'<enter>. Now type in the rest of Listing Two.

When you have finished, leave the editor, and check the procedure by typing 'OPENINDEX'. (This procedure has no parameters, so just the command is sufficient.) You will need to do this at least twice, so that you can test both branches of the procedure – once to open an existing file (eg EUCLID), and once to create a new file – (eg RECIPES). Assuming it runs correctly it will leave the file open each time, so you will also need to close it again by typing CLOSE before using it again.

The third procedure we are to add tackles the problem of getting the data into the database. To use the recipe book example, each entry, as defined by the record structure, comprises a primary word (WORD$), such as 'Jam'; an optional subsidiary word (SUBSID$), such as 'Raspberry'; and a page number (PAGE), say '17'. When we come to print the index, this would then have a general head of Jam, with a long list of all the different jams in the book, and the page they are found on. In addition, you could wish to include 'Raspberry Jam' as a 'word' in its own right. This requires a second entry, with 'Raspberry Jam' in WORD$, '17' in PAGE, and SUBSID$ left blank.

There are to be several options during data entry, and we shall use the same techniques as for OPENINDEX in finding the user selection, with one vital difference. Instead of the procedure being run through once, and then stopping, we wish to keep cycling round, getting a response, acting on it, and returning for another response. This ability of computer programs to repeat steps is another key feature that makes programs powerful and flexible. The statement in Archive that enables this is the 'WHILE' statement. It is similar to the IF statement, with a condition following it. In this case it says WHILE x is true, continue repeating a set of actions. It too has a finishing marker, 'ENDWHILE', which directs program flow back to the WHILE condition. If it is still true, the actions are repeated. If not, it goes down to below the ENDWHILE, and carries on from there.

In the case of the ENTERWORDS procedure in **Listing three**, the condition is whether the KEY$ variable is set to "S" or not. The '<>' symbol should be read as 'is not equal to', so the program will continue looping round while KEY$ is not equal to "S". Once the user enters an "S" to stop, the condition ceases to be true, and the program flow goes to below the ENDWHILE, which in this case is the end of the procedure as well.

There are a number of other points of interest in this procedure. Firstly the DISPLAY command is used. As well as showing the structure, it places the field values themselves on the screen – both

Listing Three

```
proc ENTERWORDS
  local KEY$: let KEY$=""
  display
  print at 12,5;"[N]ext [B]ack [I]nsert [A]lter [D]elete [S]top"
  while KEY$<>"S"
    sprint
    print at 12,55;"?"
    let KEY$=upper(getkey())
    print at 12,55;KEY$
    if KEY$="N": next : endif
    if KEY$="B": back : endif
    if KEY$="I": insert : endif
    if KEY$="A": alter : endif
    if KEY$="D": delete : endif
    endwhile
  endproc
```

for showing and entering the data when the appropriate commands are called. Secondly, the PRINT statement contains the phrase 'at 12,5;'. Positioning the cursor (at row 12, column 5 in this case) is neater than printing wherever the cursor happens to be. These lines are before the WHILE loop, so only get performed once when the procedure is first called. From the WHILE statement onwards, everything will be repeated every time the loop cycles round.

The 'SPRINT' command is not (alas) to make Archive go faster. It should be read as 'screen-print', and causes the contents of the DISPLAYed fields to be updated with the current record values. Having obtained the user response, the various possibilities have to be dealt with. This time, instead of an either/or situation, there are five possibilities. In this particular case, each option only requires a one word command with no 'ELSE' options. To make the layout clearer each IF...ENDIF sequence has been included entirely on one line. It is permissible in Archive to put several statements on the same line, and use colons to separate them instead of <enter>s. Another layout feature of the Archive editor is the way it uses indents, to show the procedure structure. Every time you use a command such as IF, WHILE, or CREATE, the subsequent lines are indented a couple of extra spaces, until the ENDIF, ENDWHILE, OR ENDCREATE is added.

The options themselves are some of Archive's commands for manipulating the database files. Imagine the records as card-index cards. You can change to the NEXT card or go BACK to the one before. You can make ALTERations to the current record or even DELETE it completely. (Note, this is not the same as deleting the information in each field, which results in a blank record.) And you can INSERT records. Rather like INSERT mode in the program editor, this is the main data entry option. It keeps creating more records for you to fill in, using the DISPLAY screen, until you press F4 to finish. The procedure then carries on with getting the next menu response.

Go back into the editor, and add this procedure to the two already there. If you have forgotten how, it is EDIT<enter> F3 N and start with the procedure name. To test it afterwards you will need to have an OPEN index file, for which you can use OPENINDEX. Then type ENTERWORDS. Once it is working, practice inserting a number of words, page numbers etc. You can move to the line below with either the enter key, or the tab key. If you need to go back up a line, use shift + tab. The full range of text editing keys is available.

After you have inserted a few words, press F4 to finish inserting, and try all the other options. While you are using the ENTERWORDS procedure, take the opportunity to enter some test data. This should include words with the same entries, but different pages; the same main words, but different subsidiaries, and some without subsidiaries. This will allow us to test the printing out routine later on.

We now turn to the somewhat trickier problem of getting information out of the data base; trickier because there are many more possibilities. Unlike a word-processor, a database has to be very selective in what it can print back. Just churning out a list of everything held in the file will rarely be sufficient. Different applications will have completely different requirements. In our case, we need something resembling a list. But each main word will only want printing once; followed possibly by a list of subsidiaries, each only occurring once; with perhaps several page numbers after each. This selective use of a data is vital to producing useful 'reports' from a database.

Before we look at what we print, we will look at where to print it. So far we have used the PRINT statement, which always writes on the screen. There is another command, LPRINT, which is more flexible in that it can be directed to a printer, or the screen, or to a file – for importing into Quill. **Listing four** shows the procedure that controls this. It uses the SPOOLON and SPOOLOFF commands to control where the information is sent. As with all the commands here, extra information about how they work can be gleaned from the Archive section of the manual. Enter the REDIRECT procedure now. To test it is working, type REDIRECT at the prompt, and select the [S]creen option. Test the redirection by typing LPRINT "Hello everybody" at the prompt, and it should appear on the screen. When the rest of the program is entered, and SAVEd, you could also test the other options.

Now we can go on to the procedure to print out the index, shown in **Listing five**. This is substantially more complicated than any of the other procedures, with a number of points of interest. First we set a variable called MARGIN to 10. This is not a built-in command, just a numeric variable that is used later in this procedure. Then we use the ORDER command. This is a built in command, and is one of the most powerful in Archive. It takes the data file, which has randomly entered words and numbers, and puts them all in order.

The first ordering is for WORD$, and so the records are put in alphabetical order for that field. There is a secondary order, using SUBSID$, so where two or more main words are the same, the subsidiary will be used to decide their order. And there is a third ordering, for when both WORD$ and SUBSID$ have two or more the same, using PAGE to decide the order. Using ORDER is vital to our Index program. As well as needing the words in alphabetical order, we also need to have all identical words together in a group, which is what happens when they are ORDERed.

Once the records are in order we ensure we are at the FIRST record in the file, and then start processing them. Here the WHILE loop really comes into its own. The same program lines can be used to deal with each record in turn. Our condition for looping is to keep going until the end of the file is reached. Archive provides a built in function to test this, called EOF(). We can use the NEXT command to move through the records, and EOF() will only be true once it is at the end. By using WHILE NOT EOF(), the loop will behave as required.

The complication here is that the main word wants to be printed once, and then the subsidiaries and page numbers dealt with in a loop of their own. We can actually have a second WHILE loop contained completely ('nested') within an outer one. In this case, as well as testing for the end of file, we also need to check that WORD$ hasn't changed. This is done by storing the

```
Listing four

proc REDIRECT
  print at 5,12;"[P]rinter [S]creen or [F]ile ";
  let KEY$=upper(getkey())
  spooloff
  if KEY$="S": spoolon screen : cls : endif
  if KEY$="F"
    input at 7,12;"Enter file name : ";FILE$
    spoolon FILE$ export
    endif
  endproc
```

```
Listing five

proc PRINTINDEX
  local KEY$,MARGIN,W$,S$
  let MARGIN=10
  REDIRECT
  order WORD$;A,SUBSID$;A,PAGE;A
  first
  while not eof()
    let W$=WORD$: let S$=SUBSID$
    lprint : lprint tab MARGIN;
    W$;if S$<>""
    lprint : lprint tab
    MARGIN+2;S$;
  endif
 lprint " ";PAGE;
next
while (WORD$=W$) and ( not eof())
  if (S$<>SUBSID$)
    let S$=SUBSID$
  lprint : lprint tab MARGIN+
                  2;S$;" ";PAGE;
      else
      lprint ",";PAGE;
      endif
    next
    endwhile
  endwhile
lprint : spooloff
endproc
```

original value of WORD$ in another variable (W$), and then using the comparison between them as the test. Note that the loop will only continue if test1 AND test2 are both true. If either become false, the inner loop with come to an end, and the program will continue with the outer loop. Sorting out the logic for multiple nesting and multiple tests can be one of the most baffling parts of computer programs.

The LPRINT statements handle the actual printing. Both LPRINT and PRINT can handle numerous items on a line. Variables can be mixed with text, each separated by a semi-colon. If the line ends with a semi-colon, it assumes further data is to follow for that line. If not, it ends with a carriage return/new line. A new-line can be forced by just using LPRINT on its own, with no accompanying print items. There are also TAB instructions, which indicate where to start print. We can't select rows randomly on a printer, only how far along the current row to go. In this way the MARGIN variable can be used to give a margin, adding any extra indent where required.

The strategy for the actual print lines is as follows. In the outer loop, print the main word. If the subsidiary is not blank, print that as well. The first page number for that word is also printed, before moving on to the next record. If the main word has not changed, the inner loop is entered. For each cycle of the inner loop, (in which the main word will always be identical), the subsidiary is tested. If it has changed, a new-line and new subsidiary are printed, otherwise the next page number is added to the current line. The inner loop finishes when the main word changes; and the whole lot finishes when the end of file is reached.

Type in the PRINTINDEX procedure in Listing five, paying particular attention to the punctuation marks. Errors here can drastically affect the results, and be tricky to track down. Once entered, save the program again before testing it. Longer, more complicated programs have more chance of leaving the computer 'stuck' in a loop. If the worst happens, you can then re-load it, for re-checking, and 'debugging'. When you type PRINTINDEX (and you must have an index file, already open) you should initially opt to send output to the screen. If all is well, you will see a neat, formatted index of words and page numbers, all in the correct order.

All that remains is to add a final procedure, which draws all the parts of the program together. **Listing six** does this, so enter it now. It again is simply a loop, collecting user responses and acting on them. It always starts with an OPENINDEX call, and then allows you to select ENTERWORDS, PRINTINDEX, or (after closing the current file) OPENINDEX again. Any of these can be selected in whichever order is required, and finally "S" will enable you to stop. The reason for calling it START is to enable Archive to run it automatically. Make sure you have SAVEd it before trying this. To run a program, you simply use the RUN command in place of LOAD. Then, after loading the program, Archive looks for, and runs the procedure called START. This means you don't have to remember the names of the procedures in the program. If you have designed it right, all the options can be accessible just by RUNning the program.

That concludes the INDEXER program, and also this tutorial. Hopefully you will now feel at home with entering Archive procedures and using the editor. You will also have learnt a bit about the Archive Language data files; and some of the commands and functions. The Archive reference section, in the manual, describes all of these in much more detail. The tutorial section also contains useful programming examples, and information on aspects of Archive not covered here. This should help take you further with this powerful programming environment.

USING THE ARCHIVE PROGRAM EDITOR

Whenever you need to add or alter program procedures you need to use the Archive Editor. To do this, type EDIT<enter> from the prompt.

If there are no procedures currently entered, you are only given the 'New procedure' option.

Whenever you create a new procedure, you are put automatically into INSERT LINES mode. Finish each line by pressing enter. Exit from INSERT LINES mode by pressing ESC; or by entering a blank line; or by using up or down cursor keys.

The following options are all available from the main EDIT menu. You must return to this menu before you can select a different option.

To enter INSERT LINES mode press F4. The new lines you type in will be added below the current, highlighted line.

To make alterations to an existing line use the Up or Down cursor keys, to highlight the required line, and press F5 for EDIT LINE mode. Make any necessary correction, or additions, and then press enter to finish. Alternatively, pressing ESC will abandon the edit without altering the original line.

To select a different procedure, from the list at the left of the screen, use the tab key to work down the list, or <shift>+<tab> to work up it. The highlighted procedure is then displayed in the main program area.

To erase (cut) the current, highlighted line, type <F3> C <enter>. Several lines can be highlighted for cutting, by pressing the up or down cursor keys, before pressing the enter.

The most recently cut line (or lines), can be 'pasted' back below the current, highlighted line, by typing <F3>P. As well as recovering lines cut by accident, you can use the cut and paste combination to move lines either within a procedure, or from one procedure to another.

To create a new procedure, type <F3>N. You must then enter the name you have chosen, (which can be up to 13 characters starting with a letter), plus any parameter names you require.

To delete a procedure completely, type <F3> D <enter>. A deleted procedure cannot be recovered, unless it is already stored on microdrive, in a program file.

To exit from the editor, press esc from the main EDIT menu.

Glossary of Esoteric Terms

Command - Any word Archive understands as an instruction.
Database file - collection of data records, held on microdrive/disk.
Field - single item in database record. It can be text or numeric, and is defined by the structure of the database.
Function - built in command that gives an answer back, when called.
Local Variable - one which only exists within a single procedure. Unless declared as LOCAL, a variable can be used by other procedures, even after the procedure that first used it finished.
Parameter - an item passed to a procedure when it is called, which can be used inside the parameter like a variable.
Procedure - named, self-contained program segment, that can be used like a built in command.
Record - an entry in a database file. It contains values for each of the fields for that database structure.
Report - any printed matter produced by a database program. It may be raw data, summaries, or derivations from that data.
Structure - The names and data-types of the fields that make up a record.
Variable - named memory location. If the name ends with a '$' it can store up to 255 text characters. If not, it can store a number.

Listing Six

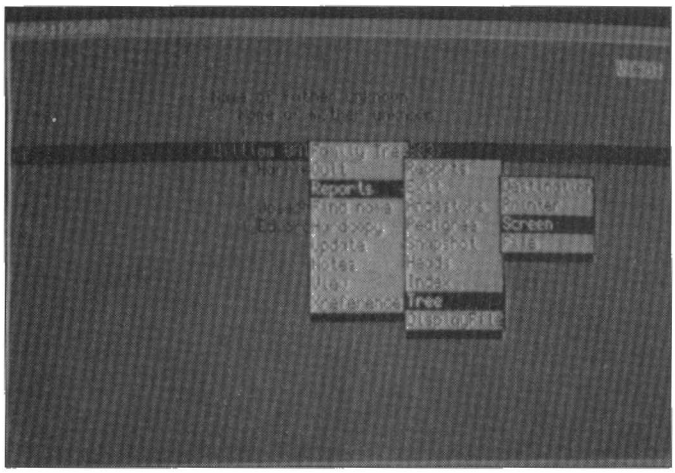
```
proc START
  local KEY$: let KEY$=""
  OPENINDEX
  while KEY$<>"S"
    cls
    print "[E]nter words, [P]rint index, [A]nother file, [S]top ";
    let KEY$=upper(getkey()): print KEY$
    if KEY$="E":ENTERWORDS: endif
    if KEY$="P":PRINTINDEX: endif
    if KEY$="A": close "INDEX":OPENINDEX: endif
  endwhile
  close "INDEX"
endproc
```

SOFTWARE FILE

QL GENEALOGIST

INFORMATION:
Program: *QL Genealogist*
Supplier: Chris Boutal, 42 Charwood Road, Wokingham, Berkshire RG11 1RY
Price: £19.50, 3.5in disk. Min. 256K memory expansion required.
A 128K cut-down version is available on mdv (if an mdv is sent with the order).

John Shaw finds a way of sorting his family out.



I have been researching my family history for some years now, and most of the data was put on Andy Carmichael's program which was available from the *QL World* Microdrive Exchange at the bargain sum of £3.00. This was very good value, but suffered the usual slowness inherent in an *Archive*-based utility and was limited to a 'one person' visual presentation.

Now, Chris Boutal has produced a family researcher's dream; he has taken eighteen months to produce a complied Superbasic program which does everything a family historian would wish to have available. Let me give you a taste of some of the features before I go into more detail.

QL_Genealogist has:
1. Rapid LOAD and SAVE times.
2. Menu driven windows.
3. Multi-tasking compatibility.
4. Time and date stamping.
5. Configuration for screen presentation and printer.
6. 'Variables' dimension control.
7. EXPORTability.
8. Total Family Tree display (on screen or printer).
9. Multiple viewing screens.
10. A notes file for each person.
11. Rapid name search.
12. Hardcopy ability on all displays.
13. Family relationship cross reference researching.
14. Ancestor lineage, alphabetically or chronologically.
15. Pedigree lineage, alphabetically or chronologically.
16. A full indexing facility.
17. A moment-in-time snapshot facility.
18. A heads of family indicator.
19. A linked research data handler.
20. A surname spelling variation file.
21. A 'getting started' tutorial chapter.
22. Example listings.
23. An example program displaying the Royal Family line since 1066.
24. A 32-page manual (A4).

I think it would be best if I go through each facility in summary form and finish off with a few tips which I found useful when entering my own family details.

Rapid LOAD and SAVE times: The 130 names I have so far entered take about 15 seconds to LOAD from the main screen which itself arrives about 10 seconds after BOOTing up. SAVEing after each session is completed in a matter of seconds.

Menu system: Menus can either be called up by pressing SPACE and moving the cursor keys or by just typing the initial letter of the selection required. There are sub-menus to widen the choice.

Multi-tasking: I found this particularly useful when transferring details from my Archive "Family Tree" program into this one. They were both multi-tasked onto *Taskmaster* and I was able to swap quite rapidly between them for reference purposes.

Time and date stamping: At the beginning of each session a Date/time sequence enables you to enter for record purposes the time and date of the data input.

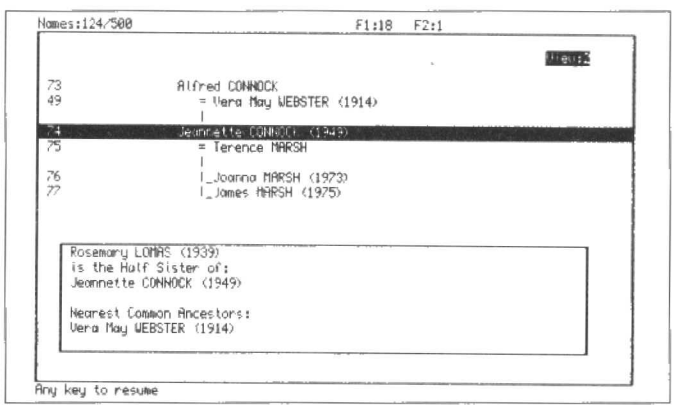
'Variables' dimension control: This caters for the number of names available for use: the default is 500 max. according to memory available. The length of characters in the surnames and forenames is also adjustable. These, of course, should be entered before the Names file is loaded. For example, the default forename parameter is 20 characters; I had to alter this to cope with one of my uncles who was christened 'Raymond William Berisford . . .', 26 including SPACES. The maximum is 40. The combined length of forename + surname must not exceed 56 characters.

Configuration: The screen colours can be altered to taste and the printer and filename defaults changed while the program is running.

Exportability: This menu option allows all the files to be exported to Archive for further processing if required.

Total family tree display: This is certainly one of the cleverest pieces of the program. Once you have decided on the starting point, the computer displays the whole family tree from that point to the present. It is shown in the usual 'tree' format and you have the option of printing it out. In the screen presentation it also has the facility to move around the tree by means of the cursor arrows.

Would it be able to deal with the ultimate test in the presentation of my wife's tree? Her father died during the War and his sister's husband married her (my wife's) own mother following a divorce! Got it? Would



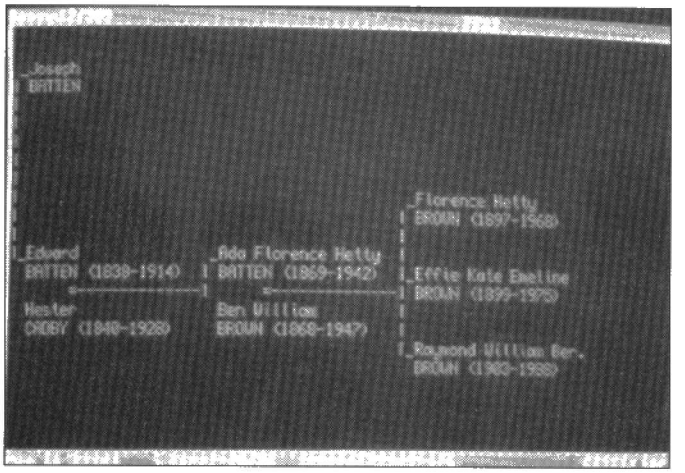
```
QL GENEALOGIST - Ancestors
Mon 27 Aug 1990 10:10

Ancestors of Angela Mary SHAW (1971)

Ada Florence Hetty BATTEN (1869-1942). Great GrandParent
Edward BATTEN (1838-1914)..... Great Great GrandParent
William BATTEN (1809-1883)..... Great (3 times) GrandParent
Jessie BLACKWELL (1894-1981)..... Great GrandParent
William BLACKWELL..... Great Great GrandParent
Anne BOWERS (1840-1924)..... Great (3 times) GrandParent
  BOWERS..... Great (4 times) GrandParent
Ben William BROWN (1868-1947)..... Great GrandParent
Effie Kate Emeline BROWN (1899-1975).. GrandParent
James BROWN..... Great Great GrandParent
Hester Millett CADBY (1840-1928)..... Great Great GrandParent
John (James) CADBY..... Great (3 times) GrandParent
Harriet HANNY..... Great (3 times) GrandParent
Norman LOMAS (1911-1944)..... GrandParent
Rosemary LOMAS (1939)..... Parent
Samuel LOMAS..... Great Great GrandParent
William Headley LOMAS (1878-1954)..... Great GrandParent
Annie PEACOCK (1880-1955)..... Great GrandParent
Albert PROCTER..... Great (3 times) GrandParent
Fanny PROCTER (1863-1932)..... Great Great GrandParent
John SHAW (1888-1950)..... Great GrandParent
John Raymond SHAW (1941)..... Parent
Samuel SHAW (1910-1964)..... GrandParent
Sarah SHEARN..... Great Great GrandParent
  WARD..... Great (4 times) GrandParent
Arthur WEBSTER..... Great Great GrandParent
John WEBSTER (1887-1950)..... Great GrandParent
Vera May WEBSTER (1914)..... GrandParent
Emma YOULE (    -1918)..... Great Great GrandParent
Mary Ann (SHAW) (1889-1955)..... Great GrandParent
```

```
QL GENEALOGIST
- Snapshot
For 1914

Mon 27 Aug 1990 10:12

Emma YOULE (    -1918)..... (
Arthur WEBSTER (    -1917)..... (
Leonard WEBSTER (    -1917)..... (
Edward BATTEN (1838-1914)..... (76)
Hester Millett CADBY (1840-1928)..... (74)
Anne BOWERS (1840-1924)..... (74)
Kate CADBY (1858-    )..... (56)
Fanny PROCTER (1863-1932)..... (51)
Frederick J BLACKMORE (1867-1951)..... (47)
Ben William BROWN (1868-1947)..... (46)
Kate M (BATTEN) (1868-1937)..... (46)
Ada Florence Hetty BATTEN (1869-1942).. (45)
Emeline BATTEN (1871-1967)..... (43)
William Headley LOMAS (1878-1954)..... (36)
Ada BLACKWELL (1878-    )..... (36)
Nelly BLACKWELL (1879-    )..... (35)
Annie PEACOCK (1880-1955)..... (34)
John WEBSTER (1887-1950)..... (27)
John SHAW (1888-1950)..... (26)
Beatrice Kate BATTEN (1889-1957)..... (25)
Mary Ann (SHAW) (1889-1955)..... (25)
Frederick Edward BLACKMORE (1894-1917). (20)
Jessie BLACKWELL (1894-1981)..... (20)
Leslie BATTEN (1896-1978)..... (18)
Herbert James BOLWELL (1896-1970)..... (18)
Florence Hetty BROWN (1897-1958)..... (17)
Gilbert BLACKMORE (1897-1970)..... (17)
Hilda Helen BEARING (1897-1974)..... (17)
Effie Kate Emeline BROWN (1899-1975)... (15)
Dora BLACKMORE (1902)..... (12)
Evelyn Maud BATTEN (1902-1922)..... (12)
Raymond William Bert BROWN (1903-1988). (11)
Grace Hilda LOMAS (1908)..... (9)
Samuel SHAW (1910-1964)..... (8)
Norman LOMAS (1911-1944)..... (7)
Vera May WEBSTER (1914)..... (6)
```

it simplify it if I told you that my wife's cousin is also her step-sister? Never mind; the important thing is, would the program cope or would it have a nervous breakdown? Of course it coped splendidly. The author had obviously considered the existence of this type of anomaly.

Multiple viewing screen: This facility works with the 'tree' display when it automatically remembers five 'views' of parts of the 'tree' and allows you to switch between them rapidly. This proved to be a very useful facility, particularly when entering a large amount of detail.

Notes file per person: This is a wonderful option which allows the display of a notes file previously entered to be displayed on the screen under the name of the person indicated.



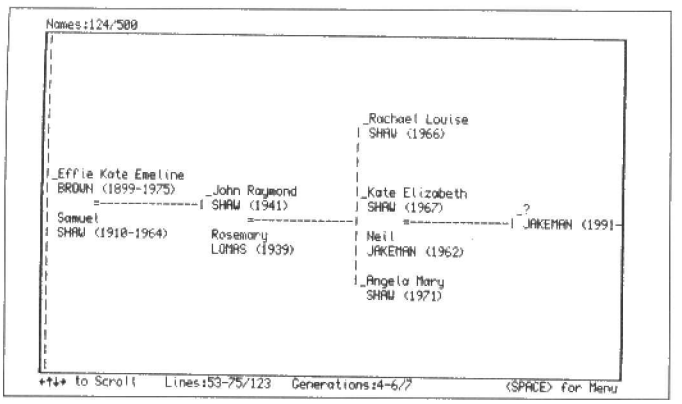
Thus, if you had your great-grandfather highlighted on the screen, by pressing ENTER all the notes you have gathered on him show up under his name. In addition, all the cross-references to him in the research file (explained later) are shown.

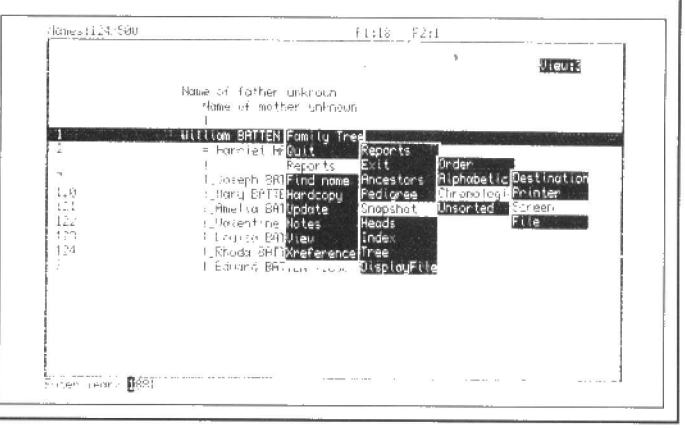
Rapid names search: Should you wish to find a particular name, or how many times a name crops up in your family history, then this section gives you all the information you require.

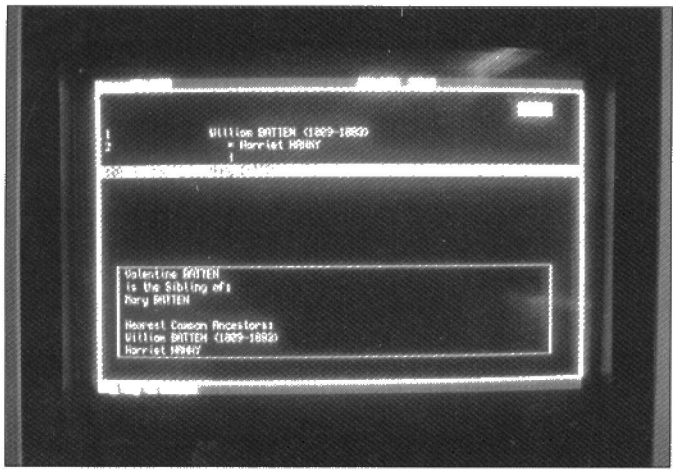
Hardcopy: Once you have configured the printer drivers, (not required for Epson compatibles), then you have a full printout ability for any of the multitude of displays.

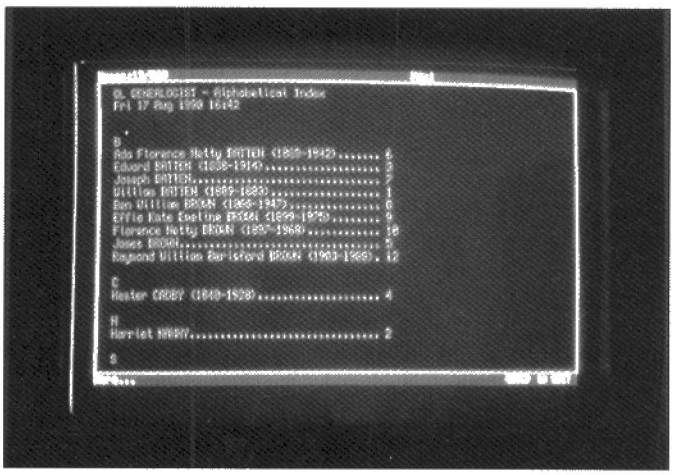
Cross reference researching: Again, this is a remarkable piece of programming which is quite outstanding in it's operation. Select any of the persons on your family tree, press the XREF bar and then select another person. A box is immediately displayed . . . "Anne Smith is the HALF SISTER of Mary Jones", or whatever the family relationship happens to be.

Ancestor and pedigree lineage: Select a starting point, for example yourself, and then via the pop up menu select either an Alphabetical or









Chronological printout of all your ancestors or your pedigree, as far back as you have programmed them. Their birth and death dates are displayed as well.

My daughter used this facility to help choose a family name for her impending baby.

Full index: This very useful screen or hardcopy printout will set out, in alphabetical sections, details of all the people you have entered into the program. Also shown will be the allocation number to enable you to cross reference them easily.

Snapshot facility: This is another quite extraordinary piece of programming. Decide on a moment in time, say 1914 (the outbreak of World War One) and the computer will give details of all your family who were alive at that time and how old they were.

Heads of family indicator: This gives a listing of all persons whom you have not allocated a Mother or Father. Thus they are at the top or head of the list. Useful to point out where your next research should start.

Linked research data handler: Separate files are created when you first set up the program, to hold details of your research, ie birth, death and marriage certificates, tombstone inscriptions, etc. When you have entered them the program allows you to cross reference each name to the one on the main file. If, for example, you have found a Golden Wedding Invitation (as I did) dated 1912, then you could enter each guest mentioned and the program searches for its counterpart in the main tree. If there is more than one with the same name it will allow you to move to the correct one. Thus, when you call up a name for study in your main file it will automatically refer to all the occurrences of the name and in which research file to find them. Remarkable!

Surname variation file: In days gone by, before compulsory education, it was quite common to find a variation in the spelling of a family name from generation to generation. Johnson became Johnstone which became Jonson, etc. The author has allowed for this and developed a file which cross relates all these occurrences for use with the IDENTITY command.

A tutorial chapter: As you will have gathered from what I have written so far, this program is detailed and complex (not complicated, I hasten to add). It was therefore useful to include a step-by-step guide to the workings of the program. You learn to create a simple, yet valuable fictitious family tree, and then the program takes you in easy stages through every aspect of data input and retrieval.

Example listings: These are included to help make sure you are on the right road in your tree building exercise.

Example program: Chris Boutal has also included a database of the Royal Family lineage since 1066 AD. This performs exactly the same as your own family history, ie you can print it out, study it or, if you think you have royal blood, add your family to it!

32-page manual: This detailed manual comes with the package and sets out all the facilities with explanations where required. It is comprehensive and obviously written by a person devoted to his hobby.

Since acquiring this program for review, I have entered over 130 different names of my ancestors. During the whole of the entry process, it behaved impeccably and was a credit to the care taken by the author to search out any bugs. I did make mistakes, but these were mainly due to inattentiveness on my part rather than difficulties with the program, which is error trapped. As an example, I married my wife to my grandfather (we both share the same name, John Shaw)! The fault, though, was highlighted in the family tree display.

Tips, well yes, I have a few. Don't forget that you need to do a backup AND run the file MAKE_DATA in order to provide the research files on flp2 (or whatever default you choose). Read the manual a few times carefully and do the exercises provided by the author, *before you start on your own tree.* Start your entries with the person FURTHEST away from you in time and then work towards yourself. The reason for this lies in the way the referencing procedures work; when you start you will have a screen in front of you which has the following headings.

This is their daughter and figures 1 and 2 point to her dad and mum. So you continue, filling in daughters and sons until you reach the present day. Then you start on another line of the family.

Make a regular printout on the index as you progress, to ensure you are getting the relationships right. Most of us have ancestors who share the same name as ourselves. Which leads me to the next tip . . .

Take your time and get some to help you and check your cross referencing. It's all too easy to get swept away with enthusiasm and try and complete everything at once. Fortunately the program is very user friendly and all mistakes can be rectified.

SAVE regularly. Sinclair's Law states that a QL will lock up 10 seconds before you intend to SAVE those three hours of inputting!

This is certainly a program written by enthusiast and, I would say, guaranteed to make even the most fastidious and exacting Family Historian go wild with delight. The whole thing is designed to introduce a newcomer to family history recording with as little pain as possible.

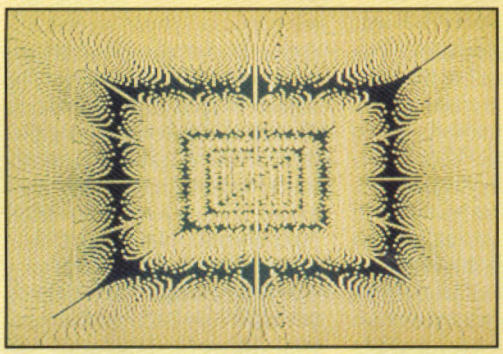
Is it good? Well, a friend of mine is going to buy a QL just to use this program . . .it's that good.

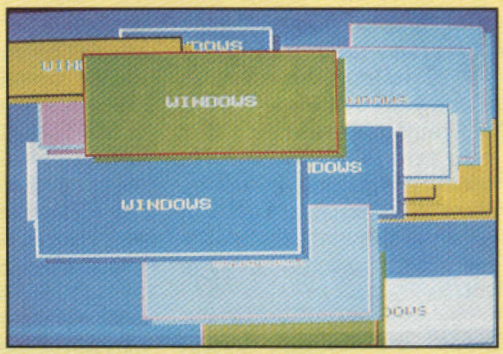
0 0 0 0
Christian name: Surname: YOB: YOD: DAD: MUM:
(YOB= year of birth . . .YOD= year of death)
You decide to input your great-great-grandparents and their children thus:
(1) Ebenezer SMITH 1815 1885 0 0
(0's are entered as you do not yet know their parents.)
(2) Jessica BROWN 1818 1879 -1 0
(-1) is entered to show she married number '1' ie Ebenezer.
(3) Effie SMITH 1840 1900 1 2

SOFTWARE FILE

QL VISION MIXER

INFORMATION:
Program: *QL Vision Mixer*
Supplier: Dilwyn Jones Computing, 41 Bro Emrys, Tal-y-Bont, Bangor, Gwynedd, Wales.
Price: £10.00, mdv, 3.5 inch or 5.25 inch disk. 256K minimum.





Have you ever admired the way in which those clever people on television can use video wipes to produce elegant and eye-catching effects, and wished that you could produce the same on your QL? Wish no more, for programming wizard Dilwyn Jones has produced a stunning compendium of more than 100 different video effects ready for you to manipulate on your computer.

Now at your fingertips are 'staggered columns', 'double wipe-ins', 'quadruple diamonds', 'four stripes up/down' and a multitude of other odd-sounding but effective screen kaleidoscopes which you can use with your graphics.

Unfortunately the wipes proved impossible to photograph in action, but we have included pictures of the sample screens.

This program comes with a very well written and easy to follow sixteen page manual and and is a must for anyone who uses their QL for displaying advertisements or information. It will also multitask under *Taskmaster*.

To start with, Dilwyn has supplied nine screen pictures which you can use to enhance the effect of your presentation. You, of course, can add your own later from such programs as *Page Designer 2*, *Desktop Publisher*, *Artice* or similar.

Using the program is simplicity itself, as Dilwyn has taken the trouble to make the set up sequence as user friendly as possible.

After the opening titles you will see the 'Screen Mode for picture display (4 or 8)?' prompt. Press either 4 or 8, depending on what mode your screen pictures were drawn in. The program has, of necessity, a blank screen built into it which can be any of the following colours: black, blue, red, magenta, green, cyan, yellow, white or random.

Having selected the colour of your choice, the program then asks you if you wish to load the screen names list, a collection of filenames of the screen pictures that you want the program to use. You can make simple screens using a screen art function in the program, but for serious work it is better to use screens from a proper computer art or DTP package. Screens from most QL art programs can be used in Mode 4 or Mode 8 (but not both together).

As the set-up sequence continues, you are invited to customise your display by a choice of finite or infinite mix sequences, random video mix effects, non-repetitive mix effects, sequential screens or random pauses. After a few minutes you have at your disposal a very professional display which will in my view stand comparison with any commercial product I have seen. Dilwyn makes no bones about it being faster on disk due to the amount of memory and loading times, but it will also work with microcassettes.

Here then we have a program, modestly priced considering the work gone into it, which will enable you to enhance your home video displays (instructions given), illustrate a lecture or provide an information display point.

```
HALF HOR L-R            HALF HOR R-L            CLOSE TOP HALF
OPEN TOP HALF           QUARTER L-R             QUARTER R-L
CLOSE QUARTERS ACROSS   OPEN QUARTERS ACROSS    BOX IN
BOX OUT                 HALF BOX IN             HALF BOX OUT
ZEBRA 1                 ZEBRA 2                 ZEBRA 3
ZEBRA 4                 ZEBRA 5                 ZEBRA 6
ZEBRA 7                 ZEBRA 8                 SIDEWAYS BOTH IN
SIDEWAYS BOTH OUT       SIDEWAYS 2 RIGHT        SIDEWAYS 2 LEFT
OPEN HALVES ACROSS      CLOSE ACROSS HALVES     WIPE DOWN
WIPE UP                 DOUBLE WIPE IN          DOUBLE WIPE OUT
OPP. VERT. HALVES 1     OPP. VERT. HALVES 2     QUARTER DOWNS
CLOSE HALVES            OPEN HALVES             QUARTER UPS
RANDOM LINES            DIAGONAL 1              DIAGONAL 2
DIAGONAL 3              DIAGONAL 4              DIAGONAL 5
DIAGONAL 6              QUARTER BOXES 1         QUARTER BOXES 2
QUARTER BOXES 3         QUARTER BOXES 4         SPIRAL IN
SPIRAL OUT              FOUR STRIPS DOWN        4 STRIPS DOWN/UP
4 STRIPS UP             4 STRIPS UP/DOWN        SILLY 1

CHOOSE THE RANGE OF EFFECTS TO USE FOR THE RANDOM SELECTION

←→↑↓=MOVE AROUND LIST ABOVE     SPACE=SELECT/DESELECT EFFECT
F1=SEE EFFECT     F2=SELECT ALL EFFECTS     ESC/ENTER=FINISH
```

SOFTWARE FILE

Aqua Vitae

INFORMATION:
Program: *Aqua Vitae*
Price: £14.95 (inc UK p+p)
256K minimum.
Suppliers: Qlympic Computer Software, Quellenweg 18, 4220 Dinslaken, Germany.

John Shaw plunges into a new adventure with his eyes open.

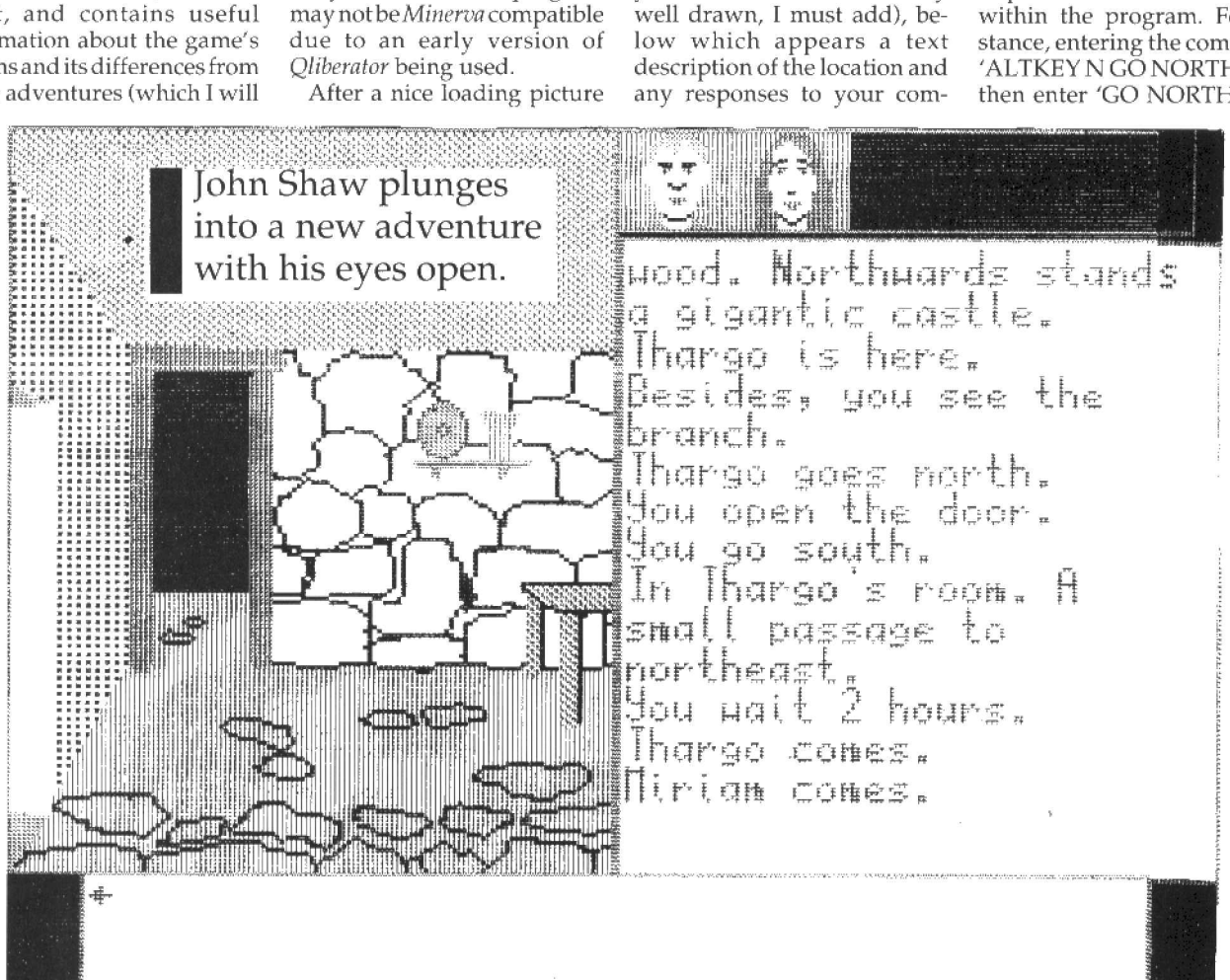
Waking up from a dream about a far distant country, in which you have a quest to finish, you decide to forget about your hero's life, and set off in search of new adventure. So much the manual tells you but as to what that quest is or how you are to finish it is left up to you to discover.

The manual is a Quill document, and contains useful information about the game's origins and its differences from other adventures (which I will attempt to illustrate later). The manual contains a full list of all the verbs used by the game, but this does not make the game any easier — it just cuts down the frustration of trying to work out just what the game will understand. So, enough of the manual and onto the game.

Just a quick mention that the early versions of the program may not be *Minerva* compatible due to an early version of *Qliberator* being used.

After a nice loading picture and some soothing music, the main adventure loads. The screen is split into four parts: the left is taken up with a picture of your current location, while below this is the command line where you enter your commands. On the right appear small pictures of any characters whom you meet on your travels (which are very well drawn, I must add), below which appears a text description of the location and any responses to your commands. The author (Uwe Raabe) has used a special device driver for the text output which makes the text slightly smaller than the usual mode 8 characters, and also uses a very nice font. On early versions, this device driver may not actually work, causing the text output to be very badly displayed.

Before I comment on the actual game, I must first explain about the command entry system. All commands must unfortunately be typed in full ('GO NORTH' instead of the more usual 'N' for instance), but this is somewhat alleviated by the useful implementation of ALTKEYS within the program. For instance, entering the command 'ALTKEY N GO NORTH' will then enter 'GO NORTH' into



the command line whenever you press <ALT><N>. With 26 letters to use in this way, you can make typing a whole lot easier than on a normal adventure. What is more, when you save your position, these definitions are also stored so that they will be available when you later return to the game.

Each location in the game has its own picture which is loaded off disk. Although this slows down the program a little, it is nice to know that the game will also continue quite happily should you accidentally remove the disk (it issues a warning that the picture file cannot be found). The pictures themselves are well drawn; they are in fact an important part of the adventure, since some objects appear in the picture but not in the text, even though you need to do things with them!

Unfortunately the English text suffers a little in translation from the German original, but it is not so bad as to be frustrating. Although the program's vocabulary is not very large (for instance only TAKE is allowed, not GET) the manual does give a list of all of the words understood by the program, which certainly helps with the few non-standard verbs used by his program: for example, LIST is used instead of the more normal command INVENTORY.

Indeed the parser is actually more complex than would at first appear, since you can string together more than one command with 'AND'; also , if the second verb is the same, there is no need to repeat it. For instance the program would understand 'GO NORTH AND TAKE ROPE AND TORCH'. It is touches like this that give the game its character, rather than the few minor mistakes in the translation.

The game soon becomes engrossing and you begin to forget these little idiosyncracies as you begin to feel tired, hungry and thirsty and must start looking for food and drink. There is not too much food scattered about so you mus be careful how much you eat. However, the game does supply methods of saving your energy — you can REST for a given length of time, or SNEAK about (as compared to RUN if you are feeling energetic). Thankfully, you can progress quite far in the game before any of this becomes a problem, but if you do feel in need of relaxation, when you die, you appear in another location (Heaven), which has a harp to sooth away your nerves (pity I haven't found any way to be resurrected).

As you move around, you meet different characters, who will try to help you if you can give them the right objects. It is quite easy to converse with them; for example, you can 'ASK MIRIAM ABOUT THE RING', and if she can, she will reply to your question. This is essential since the computer will not give you any help upon your quest; instead you rely upon the assistance of those whom you meet.

Although there does not appear to be many locations at first, you will find that opening a few things or lifting them will reveal other objects or means of exit. There are nice touches here in that SEARCHing an object will automatically place anything you find in your hand; and entering the command GO AROUND will try all possible directions to move until it finds you an exit (if there is one).

An odd means of getting around is to FOLLOW one of the characters whom you meet, although this can cause difficulties in that you will soon become disorientated as they seeminly move around in circles using hidden passages between locations which you cannot use on your own. Still, this may be useful in a later stage of the game if you get lost somewhere.

Overall the adventure makes a refreshing change in presentation, and bodes well for future releases by Qlympic. Although there are not all that many locations in the adventure (around 30), it will certainly keep you going for many hours, since the solution to the game relies on both your reading skill and your observational ability.

QL SUPERTOOL KIT II by Tony Tebby

Over 118 Commands:- Full Screen Editor, Key Define Print Using, Last Line Recall, Altkey, Job Control, File Handling, Default Directories, Extended Network,
16k Eprom Cartridge Version ..... @£24.15d
Configurable Version on Microdrive ..... @£24.15d

MIRACLE SYSTEMS PRODUCTS

DK Disc Interface (Inc Tool Kit II) ..... @£99.82b
QL Centronics Printer Interface ..... @£28.75d
QL Expanderam 512K Thrucard ..... @£99.99e

QL HARDWARE

Single 3.5" Disc Drive & (Own PSU) ..... @£103.50a
Dual 3.5" Disc Drive & (Own PSU) ..... @£188.60a
3.5" DS/DD Discs - 10 off ..... @£9.20c
Q POWER REG. The only real solution to you QL overheating (switched mode power supply run cold) £24.15c
QL Keyboard Membrane ..... @£11.50d
QEP III Advanced Eprom Programmer ..... @£121.90d
Care Eprom Cartridges each ..... @5.75c
ULA CHip ZX6301 ..... @£15.64c

PRINTERS

STAR LC 10 Mono ..... @£179.40a
STAR LC 10 Colour ..... @£224.25a
CITIZEN SWIFT 24-pin Colour ..... @£356.50a

SOFTWARE 87 (State MDV or Disc)

TEXT87V.300 (Requires Min 256k) ..... @£60.00d
FOUNTED 89 ..... @£15.00c
FOUNTEXT 88 ..... @£25.00c
TEXT 87/FOUNTED 89/FOUNTEXT 88 ..... @94.99b
248B PRINTER DRIVER ..... @£15.00c
Upgrade to Text 87 V.3.00. Return old copy together with ..... @£25.00d

MONITORS (Price including lead)

Philips BM7522 Amber Hi-Res ..... @£97.75a
Philips CM6833 Colour Med-Res ..... @£253.00a
Philips AV7300 TV/tuner for above ..... @£69.00b
Philips BM7502 Green Hi-Res £97.76a

TONY TEBBY SOFTWARE (QJUMP)

QLFP (Micro/P disk interface upgrade) ..... @£14.95d
QMON II ..... @£23.00d
QTYP Type/Spell Checker ..... @£29.09d
QPAC 1-Desk top accessories, calander, alarm, calculator, typewriter, digital clock sysmon ..... @£21.85d

ZITASOFT SOFTWARE by Steve Jones

LOCKSMITHE copies M/DRIVE - M/DRIVE .... @£14.95c
4MATTER + LOCKSMITHE copies M/DRIVE - DISC ..... @£23.00c

The above programs are not for use in the UK

SIDEWINDER - High resolution printer driver prints full screens or parts of screens from postage stamp size to large banners. Prints sideways, invert, scale, mirror, text insertion ..... @£19.95c
SIDEWINDER PLUS - (for expanded QL's) includes all the features of above, PLUS multiple label printing, desktop publishing files and printer driver for 24" pin plus LC10 and Jx80 colour printers. (Please state 3.5" disc or M/D) ..... @£23.00c
Upgrade to Sidewinder Plus ..... @£8.05c

HEAT TRANSFER RIBBONS & PENS

Print your own T-shirt Design. Just print on paper and iron onto your T-shirt.
Star LC10NX1000, Rainbow ..... @£17.25c
Star LC10NX1000, Black ..... @£11.50c
Epson FX80, MX80, LX80, FX100/Okidata ML80 Citizen 120D/Star NX10, Black, Panasonic 1080/81 ... @£11.50c
Small 5 colour pen set ..... @£12.65c
Jumbo 5 Colour Pen Set ..... @£16.10c

READYMADE LEADS

RGB QL to Phono ..... @£5.75c
RGB 8-6 pin DIN ..... @£7.13c
RGB 8-7 pin DIN (Hitachi) ..... @£7.13c
RGB 8-7 pin DIN (Ferguson) ..... @£7.13c
RGB 8 pin to SCART (Euro) ..... @£11.50c
6-way PCC 25-way 'D' (Printer-Ser 1) ..... @£9.89c

QPAC II

QPAC II Main menu windows adjust automatically to size. Files, directory, view, print, delete, backup, jobs, pick, Rjob, sort, channels, things exec, wake, buttons, Hotkey, Hotjobs. Fully multitasking, allows many programs to run at once. Requires min of 256k expanded memory ..... @£49.91b
Upgrade QRAM to QPAC II return master ..... @£29.90b

AS REVIEWED QL WORLD AUGUST ISSUE

CARE ELECTRONICS

OPEN
9am-5pm Mon-Thu
9am-4pm Fri

800 ST ALBANS ROAD,
GARSTON, WATFORD,
HERTS, WD2 6NL.
Tel: 0923-672102
Fax: 0923-662304

Please add carriage
a=£11.50 b=£3.45
c=£1.38 d=2.30

HOW TO ORDER: ALL PRICES INCLUDE VAT

By Post. Enclose your Cheque/PO made payable to CARE Electronics or use ACCESS/VISA. Allow 7 days for delivery

DIGITAL PRECISION LTD

SUPERCHARGE SUPERBASIC COMPILER

NEW!

★ This amazing compiler translates QL SuperBasic into 68008 machine code, completely automically! It supports the entire syntax of SuperBasic (except, obviously, commands like EDIT,LIST,RENUM which are specific to Basic and would be meaningless in machine code). That means all the handling, multidimensional arrays, procedures (with parameters too), local variables, strings, sound, graphics, separate superspeed integer, floating point arithmetic (displaying Nine Digits of Precision versus The Interpreter's Seven!), extensions to Basic: the complete range of SuperBasic control constructs. ... refer to your QL User Guide for the whole list!

★ The code generated by Supercharge runs incredibly fast ... here are the standard PCW benchmarks (PCW June 1984) for QL Basic compared with the timings for the same programs when Supercharged:

BENCHMARK NUMBER	1	2	3	4	5	6	7	8
SUPERBASIC (SECS)	2.1	6.4	10.7	10.3	13.2	26.1	61.8	25.8
SUPERCHARGE (Floating Point) (SECS)	**0.2**	**0.3**	**1.2**	**0.9**	**1.0**	**2.5**	**4.1**	**8.6**
SUPERCHARGE (Integer) (SECS)	**0.06**	**0.1**	**0.3**	**0.3**	**0.3**	**0.7**	**1.0**	**N/A**

(Variations will exist between different QL systems)

Hence on standard benchmarks a 60 × speed increase is possible ... SUPERCHARGE, however, is at its least impressive when compiling such short programs, as SuperBasic grows slower as program size increases. Speed improvements of **SEVERAL HUNDRED TIMES** are in no way uncommon when using SUPERCHARGE on programs of average size.

★ SUPERCHARGED programs are independent jobs (so you don't need to have SUPERCHARGE loaded to run them).
★ SUPERCHARGED programs are relocatable and are inherently multitasking (so you can run any number of them at once).
★ SUPERCHARGE optimises as it compiles – and the advanced user may switch the optimisation from time to space on a line by line basis.
★ SUPERCHARGED programs load far more quickly (at least 5-10 times) and, hence reliably, than normal SuperBasic – there is no pause between blocks.
★ SUPERCHARGE can compile programs of ANY size, provided you have enough RAM on board (for the unexpanded QL the limiting source code size is still a massive 40K!). Compilation listings can be directed to any device – and a complete set of compile-run time error messages (clear and unambiguous, unlike the Interpreter's) are issued, showing exact error positions.
★ Add-on commands, functions and procedures are allowed by SUPERCHARGE, provided they do not read/alter Interpreter data structures ... Hence utilities supplied with disk systems and QL Toolkits will work with SUPERCHARGE!
★ SUPERCHARGE is compatible with all disk and memory-expansion systems. The system we recommend (on grounds of quality, speed and reliability) is the one we used in developing SUPERCHARGE – the CST disk system, available from Computamate (Scotia Road, Burslem, Stoke-on-Trent – Tel: 0782 811711) or from us – write in for details.
★ SUPERCHARGED programs are protected against unauthorised modification/listing/tampering ... permanently.
★ SUPERCHARGE is the only QL Compiler which offers the user the convenience of interactive testing, editing and debugging.
★ SUPERCHARGE cures most bugs and lifts many of the restrictions imposed by the Interpreter! Examples of problems completely cured under SUPERCHARGE: "Too many parameters in a procedure" bug, the RESPR bug, GOSUB in a single line FOR statement, CALL in long programs, procedure parameters/integers/string variables used in SELECT statements, full integer arithmetic, etc, etc.
★ SUPERCHARGE adds a number of new commands to SuperBasic, including ones to monitor device statuses, to list/stop/remove tasks, to change task priorities, and so on.
★ SUPERCHARGE is supplied either on disk or microcartridge and can be easily transferred/backed up from one to the other.
★ SUPERCHARGE is supplied with a superb Instruction Manual (40,000 words!) designed to complement the User Guide ... it is designed for both beginner and advanced user, with sections on "Getting The Best From SUPERCHARGE" and many examples.
★ SUPERCHARGE itself loads and runs phenomenally fast – yes, SUPERCHARGE is itself a SUPERCHARGED program!

£59.95 COMPLETE WITH DETAILED MANUAL (specify cartridge or CST diskette)

AVAILABLE NOW!

Author: Simon Goodwin. Assistant: Gerry Jackson. Mission Control: Freddy Vachha.

NOTE: Software houses requiring to use SUPERCHARGE will need a site licence, for which the VAT-inclusive cost is £250 (in addition to the price of one compiler). Instructions are supplied to site licence holders on the making of appropriate numbers of copies of SUPERCHARGE for their exclusive use. Site licence holders are exempted from paying Digital Precision royalties on SUPERCHARGED programs.

Why use C or Pascal when a highly structured, full-featured Basic of superior speed (once SUPERCHARGED) and flexibility is already on your QL? SUPERCHARGE is the final solution. SUPERCHARGE all your Basic programs (past, present and future) – including any you have bought – and realise your potential as well as that of your superb computer.

Digital Precision Software is available wholesale from TBD, Lightning, Websters and Microdea, or retail from selected branches of W. H. Smiths, Boots, Menzies, Computers of Wigmore Street and other fine retailers. The quickest way to order is directly from us – why not use the coupon below? Our delivery is by first class return of post. Dealer enquiries are most welcome (our discounts are generous!).

To: DIGITAL PRECISION LTD

Please rush me a copy of SUPERCHARGE, The State of the Art SuperBasic Compiler immediately. I enclose a cheque/postal order/cash for £59.95 (for orders from Europe add £2.50. Airmail orders from other countries add £4). For all orders from abroad use Eurocheques or cheques drawn on UK banks only. Within the UK our deliveries are post free.

Name. ... ...